MBS Mac64bit Plugin Documentation

Christian Schmitz

March 10, 2024

0.1 Introduction

This is the PDF version of the documentation for the Xojo Plug-in from Monkeybread Software Germany.
Plugin part: MBS Mac64bit Plugin

0.2 Content

Chapter 1

List of Topics

Chapter 2

List of all classes

Chapter 3

List of all controls

Chapter 4

List of all modules

Chapter 5

Calendar

5.1 class EKAlarmMBS

5.1.1 class EKAlarmMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An EKAlarm object represents an alarm in Event Kit.
Notes: Use the alarmWithAbsoluteDate: and alarmWithRelativeOffset class methods to create an alarm and use the properties to set information about an alarm. In OS X Mountain Lion, you can specify an action to trigger when the alarm fires via the emailAddress, soundName, or url property.
Subclass of the EKObjectMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.5pr6

5.1.2 Methods

5.1.3 alarmWithAbsoluteDate(d as date) as EKAlarmMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an alarm with an absolute date.

5.1.4 alarmWithAbsoluteDateTime(d as dateTime) as EKAlarmMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an alarm with an absolute date.

5.1.5 alarmWithRelativeOffset(offset as Double) as EKAlarmMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an alarm with a relative offset.
Notes: offset: The offset from the start of an event, at which the alarm fires.

Negative offset values fire before the start of the event, while positive values fire after the start.

5.1.6 Constructor(date as date)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an alarm with an absolute date.
See also:

- 5.1.7 Constructor(date as dateTime) 140
- 5.1.8 Constructor(offset as Double) 140

5.1.7 Constructor(date as dateTime)

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an alarm with an absolute date.
See also:

- 5.1.6 Constructor(date as date) 140
- 5.1.8 Constructor(offset as Double) 140

5.1.8 Constructor(offset as Double)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an alarm with a relative offset.
Notes: offset: The offset from the start of an event, at which the alarm fires.

Negative offset values fire before the start of the event, while positive values fire after the start.
See also:

5.1.9 copy as EKAlarmMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of an alarm.

5.1.10 Properties

5.1.11 absoluteDate as date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The absolute date for the alarm.
Notes: If you set this property for a relative offset alarm, it loses the relative offset and becomes an absolute alarm.
(Read and Write property)

5.1.12 absoluteDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The absolute date for the alarm.
Notes: If you set this property for a relative offset alarm, it loses the relative offset and becomes an absolute alarm.
(Read and Write property)

5.1.13 emailAddress as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The recipient of an email to send when the alarm triggers.
Notes: Assigning this property a value will set the soundName and url properties to nil.
(Read and Write property)

5.1.14 proximity as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A value indicating how a location-based alarm is triggered.
Notes: Alarms can be set to trigger when entering or exiting a location specified by structuredLocation. By default, alarms are not affected by location.
(Read and Write property)

5.1.15 relativeOffset as Double

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The offset from the start of an event, at which the alarm fires.
Notes: If you set this value for an absolute alarm, it loses its absolute date and becomes a relative offset alarm.
(Read and Write property)

5.1.16 soundName as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The name of the sound to play when the alarm triggers.
Notes: The value of this property is the name of a system sound that can be used with the soundNamed class method to create an NSSound object. Assigning this property a value will set the emailAddress and url properties to nil.
(Read and Write property)

5.1.17 structuredLocation as EKStructuredLocationMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The location to trigger an alarm.
Notes: This property is used in conjunction with proximity to perform geofence-based triggering of reminders.
(Read and Write property)

5.1.18 type as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The type of action to trigger when the alarm fires.
Notes: To set the type of alarm, define one of emailAddress, soundName, or url.
(Read only property)

5.1.19 url as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The URL to open when the alarm triggers.
Notes: Assigning this property a value will set the emailAddress and soundName properties to nil.

Available in OS X v10.8 and later.
Deprecated in OS X v10.9.
(Read and Write property)

5.1.20 Constants

Proximity Values

Constant	Value	Description
kProximityEnter	1	The alarm is set to fire when entering a region.
kProximityLeave	2	The alarm is set to fire when leaving a region.
kProximityNone	0	The alarm has no proximity trigger.

Types

Constant	Value	Description
kTypeAudio	1	Play audio
kTypeDisplay	0	Display alert
kTypeEmail	3	Email
kTypeProcedure	2	Procedure

5.2 class EKCalendarItemMBS

5.2.1 class EKCalendarItemMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The class for a calendar item.
Notes: The EKCalendarItem class is a an abstract superclass for calendar events and reminders. This class provides common properties and methods for accessing properties of calendar items such as the ability to set the calendar, title, and location as well as support for attaching notes, displaying attendees, setting multiple alarms, and specifying recurrence rules.
Subclass of the EKObjectMBS class.
This is a subclass of an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.2.2 Methods

5.2.3 addAlarm(alarm as EKAlarmMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an alarm to the receiver.

5.2.4 addRecurrenceRule(rule as EKRecurrenceRuleMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a recurrence rule to the recurrence rule array.
Notes: The implementation only supports a single recurrence rule. Adding a recurrence rule replaces the single recurrence rule.

5.2.5 alarms as EKAlarmMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The alarms associated with the calendar item, as an array of EKAlarm objects.
Notes: This property is empty if the calendar item has no alarms.

5.2.6 attendees as EKParticipantMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The attendees associated with the calendar item, as an array of EKParticipant objects.
Notes: This property is read-only; it is not possible to add attendees with Event Kit. This property is nil if the calendar item has no attendees.

5.2.7 recurrenceRules as EKRecurrenceRuleMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The recurrence rules for the calendar item.
Notes: The implementation only supports a single recurrence rule.

5.2.8 removeAlarm(alarm as EKAlarmMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes an alarm from the calendar item.

5.2.9 removeRecurrenceRule(rule as EKRecurrenceRuleMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a recurrence rule from the recurrence rule array.
Notes: The implementation only supports a single recurrence rule.

5.2.10 setAlarms(alarms() as EKAlarmMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the alarms array.

5.2.11 setRecurrenceRules(rules() as EKRecurrenceRuleMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the recurrences rules array.

5.2.12 Properties

5.2.13 calendar as EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar for the calendar item.
Notes: (Read and Write property)

5.2.14 calendarItemExternalIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar item,Äôs external identifier as provided by the calendar server.
Notes: This identifier allows you to access the same event or reminder across multiple devices.

There are some cases where duplicate copies of a calendar item can exist in the same database:

- A calendar item was imported from an ICS file into multiple calendars
- An event was created in a calendar shared with the user and the user was also invited to the event
- The user is a delegate of a calendar that also has this event
- A subscribed calendar was added to multiple accounts

In such cases, you should choose between calendar items based on other factors, such as the calendar or source.

Recurring event identifiers are the same for all occurrences. If you wish to differentiate between occurrences, you may want to use the start date.

For Exchange servers, the identifier is different between iOS and OS X and different between devices for reminders.
(Read only property)

5.2.15 calendarItemIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar item,Äôs unique identifier.
Notes: This property is set when the calendar item is created and can be used as a local identifier. Use calendarItemWithIdentifier to look up the item by this value.

A full sync with the calendar will lose this identifier. You should have a plan for dealing with a calendar whose identifier is no longer fetch-able by caching its other properties.
(Read only property)

5.2.16 creationDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The date that this calendar item was created.
Notes: If nil, this property was not set or was synced in this state.
(Read only property)

5.2.17 creationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The date that this calendar item was created.
Notes: If nil, this property was not set or was synced in this state.
(Read only property)

5.2.18 hasAlarms as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the calendar item has alarms.
Notes: If true, the calendar item has alarms; otherwise it does not.
(Read only property)

5.2.19 hasAttendees as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the calendar item has attendees.
Notes: If true, the calendar item has attendees; otherwise it does not.
(Read only property)

5.2.20 hasNotes as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the calendar item has notes.
Notes: If true, the calendar item has notes; otherwise it does not.
(Read only property)

5.2.21 hasRecurrenceRules as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the calendar item has recurrence rules.
Notes: If true, the calendar item has recurrence rules; otherwise it does not.

The implementation only supports a single recurrence rule. Adding a recurrence rule replaces the single recurrence rule.
(Read only property)

5.2.22 lastModifiedDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The date that the calendar item was last modified.
Notes: (Read only property)

5.2.23 lastModifiedDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The date that the calendar item was last modified.
Notes: (Read only property)

5.2.24 location as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The location associated with the calendar item.
Notes: This property is "" if the calendar item has no location.
(Read and Write property)

5.2.25 notes as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The notes associated with the calendar item.
Notes: (Read and Write property)

5.2.26 timeZone as NSTimeZoneMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The time zone for the calendar item.
Notes: If nil, the calendar item is a floating event. A floating event is not tied to a particular time zone. It occurs at a given time regardless of the time zone —for example, ,Äúlunch at noon.,Äù The start and end times of a floating event should be set as if they were in the system time zone.
(Read and Write property)

5.2.27 title as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The title for the calendar item.
Notes: (Read and Write property)

5.2.28 URL as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The URL for the calendar item.
Notes: (Read and Write property)

5.3 class EKCalendarMBS

5.3.1 class EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An instance of the EKCalendar class represents a calendar in Event Kit.
Notes: Use the properties in this class to get attributes about a calendar, such as its title and type. Use the calendarForEntityType method to create a calendar object.
Subclass of the EKObjectMBS class.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- MBS Xojo Plugins, version 19.5pr1

5.3.2 Methods

5.3.3 calendarForEntityType(entityType as Integer, eventStore as EKEventStoreMBS) as EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new calendar that can contain the given entity type.
Notes: entityType: The entity type that this calendar may support.
eventStore: The event store in which to create this calendar.

Returns the created calendar.

You can only create calendars that accept either reminders or events. Some servers might allow mixing the two, although it is not common.

5.3.4 Constructor(entityType as Integer, eventStore as EKEventStoreMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new calendar that can contain the given entity type.
Notes: entityType: The entity type that this calendar may support.
eventStore: The event store in which to create this calendar.

Returns the created calendar.

You can only create calendars that accept either reminders or events. Some servers might allow mixing the two, although it is not common.

5.3.5 Properties

5.3.6 allowedEntityTypes as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The entity types this calendar can contain.
Notes: While Event Kit only allows creation of single-entity calendars, other servers might allow mixed-entity calendars.
(Read only property)

5.3.7 allowsContentModifications as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether you can add, edit, and delete items in the calendar.
Notes: (Read only property)

5.3.8 calendarIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A unique identifier for the calendar.
Notes: This property is set when the calendar is created and can be used as a local identifier. Use calendarWithIdentifier to get a calendar with the specified identifier.

A full sync with the calendar will lose this identifier. You should have a plan for dealing with a calendar whose identifier is no longer fetch-able by caching its other properties.
(Read only property)

5.3.9 CGColor as Variant

Plugin Version: 21.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The calendar,Äôs color.
Notes: This property is the equivalent of the color property in macOS.
(Read and Write property)

5.3.10 color as NSColorMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar,Äôs color.
Notes: (Read and Write property)

5.3.11 Immutable as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the calendar,Äôs properties can be edited or deleted.
Notes: If true, the calendar is immutable; otherwise it is not. Events and reminders can still be added to an immutable calendar.
(Read only property)

5.3.12 source as EKSourceMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The source object representing the account to which this calendar belongs.
Notes: This property can only be set for newly created calendar objects. This property is read-only after the first time it is set; setting a value to this property after the first time it is set will result in an error. Therefore, moving a calendar from one source to another is not supported.
(Read and Write property)

5.3.13 Subscribed as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the calendar is a subscribed calendar.
Notes: If true, the calendar is a subscribed calendar; otherwise it is not.
(Read only property)

5.3.14 supportedEventAvailabilities as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The event availability settings supported by this calendar, as indicated by a bitmask.
Notes: If the calendar doesn,Äôt support event availability settings, this value is EKCalendarEventAvailabilityNone.

(Read only property)

5.3.15 title as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar,Äôs title.
Notes: (Read and Write property)

5.3.16 type as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar,Äôs type.
Notes: CalDAV-subscribed calendars are of type EKCalendarTypeCalDAV with Subscribed set to true.
(Read only property)

5.3.17 Constants

Entity Mask

Constant	Value	Description
kEntityMaskEvent	1	Event
kEntityMaskReminder	2	Reminder

Types

Constant	Value	Description
kEntityTypeEvent	0	Entity Event
kEntityTypeReminder	1	Entity Reminder

Event Availabilities

Constant	Value	Description
kEventAvailabilityBusy	1	Busy
kEventAvailabilityFree	2	Free
kEventAvailabilityNone	0	Not available
kEventAvailabilityTentative	4	Tentative
kEventAvailabilityUnavailable	8	Unavailable

Calendar Types

Constant	Value	Description
kTypeBirthday	4	A birthday calendar.
kTypeCalDAV	1	A CalDAV or iCloud calendar.
kTypeExchange	2	An Exchange calendar.
kTypeLocal	0	A local calendar.
kTypeSubscription	3	A locally subscribed calendar.

5.4 class EKEventMBS

5.4.1 class EKEventMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An instance of the EKEvent class represents an event added to a calendar in the Event Kit framework.
Notes: Use the eventWithEventStore method to create a new event. Use the properties in the class to get and modify certain information about an event. Other properties, such as the event,Äôs title and calendar, are inherited from the parent class EKCalendarItem.
Subclass of the EKCalendarItemMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 22.0
- MBS Xojo Plugins, version 22.0pr7
- MBS Xojo Plugins, version 20.5pr6
- MBS Xojo Plugins, version 19.4pr7

Xojo Developer Magazine

- 21.1, page 29: News from MBS Xojo Plugins, What's up with MonkeyBread Software by Stefanie Juchmes

5.4.2 Methods

5.4.3 compareStartDateWithEvent(other as EKEventMBS) as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Compares the start date of the receiving event with the start date of another event.
Notes: other: The event to compare against.

NSOrderedAscending	-1	if the start date of the receiver precedes the start date of other.
NSOrderedSame	0	if the start dates of the two events are identical.
NSOrderedDescending	1	if the start date of the receiver comes after the start date of other.

5.4.4 Constructor(eventStore as EKEventStoreMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a new event belonging to a specified event store.
Notes: eventStore: The event store to which the event belongs.

5.4.5 eventWithEventStore(eventStore as EKEventStoreMBS) as EKEventMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a new event belonging to a specified event store.
Notes: eventStore: The event store to which the event belongs.

5.4.6 refresh as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Updates the event‚Äôs data with the current information in the Calendar database.
Notes: If the event was successfully refreshed, true; otherwise, false.

You should call this method only on events that your application is editing, and only when your application receives the EKEventStoreChangedNotification notification. If this method returns false, the event has been deleted or otherwise invalidated, and you should not continue to use it.

This method does not replace the values of any properties that you have modified.

5.4.7 setEndDate(d as date, tz as NSTimeZoneMBS)

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: Sets the end date with given time zone.

5.4.8 setStartDate(d as date, tz as NSTimeZoneMBS)

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: Sets the start date with given time zone.
Example:

```
Dim es As New EKEventStoreMBS
Dim e As New EKEventMBS(es)

Dim d As New date
```

```
d.day = 2
d.Month = d.Month + 6

// sets with time zone in the date object
e.startDate = d

Dim d1 As date = e.startDate

// sets with the time zone local to that date for current user
e.setStartDate(d, NSTimeZoneMBS.localTimeZone)

Dim d2 As date = e.startDate


MsgBox d1.SQLDateTime + EndOfLine + d2.SQLDateTime

Break // you may see a difference due to summer/winter time switching
```

5.4.9 Properties

5.4.10 AllDay as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the event is an all-day event.
Notes: (Read and Write property)

5.4.11 availability as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The availability setting for the event.
Notes: This setting is used by CalDAV and Exchange servers to indicate how the event should be treated for scheduling purposes.

If the event,Äôs calendar does not support availability settings, this property,Äôs value is EKEventAvailabilityNotSupported.
(Read and Write property)

5.4.12 birthdayContactIdentifier as String

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: The contact identifier of the person for this birthday event.
Notes: This property only applies to events in the built-in Birthdays calendar. It specifies the contact identifier (for use with the Contacts classes) of the person for whom the system created this event. For any other type of event, this property returns nil.
Requires macOS 10.11 or newer.
(Read only property)

5.4.13 birthdayPersonUniqueID as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The Address Book framework record identifier of the person for this birthday event.
Notes: This property is only set if this is a birthday event; otherwise the property is "".
(Read only property)

5.4.14 endDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The end date for the event.
Notes: (Read and Write property)

5.4.15 endDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The end date for the event.
Notes: (Read and Write property)

5.4.16 eventIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A unique identifier for the event.
Notes: You can use this identifier to look up an event with the EKEventStore method eventWithIdentifier. If the calendar of an event changes, its identifier most likely changes as well.
(Read only property)

5.4.17 isDetached as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether an event is a detached instance of a repeating event.
Notes: This value is true if and only if the event is part of a repeating event and one or more of its attributes have been modified from the repeating event,Äôs default attributes.
(Read only property)

5.4.18 occurrenceDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The original occurrence date of an event if it is part of a recurring series.
Notes: This value will remain the same even if the event has been detached and its start date has changed. Floating events (such as all-day events) are returned in the default time zone.
(Read only property)

5.4.19 occurrenceDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The original occurrence date of an event if it is part of a recurring series.
Notes: This value will remain the same even if the event has been detached and its start date has changed. Floating events (such as all-day events) are returned in the default time zone.
(Read only property)

5.4.20 organizer as EKParticipantMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The organizer associated with the event.
Notes: This property is nil if the event has no organizer.
(Read only property)

5.4.21 startDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The start date of the event.
Notes: Floating events such as all-day events are returned in the default time zone.

(Read and Write property)

5.4.22 startDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The start date of the event.
Notes: Floating events such as all-day events are returned in the default time zone.
(Read and Write property)

5.4.23 status as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The status of the event.
Notes: You should act based on an event,Äôs status only if the status is EKEventStatusCanceled, which indicates that the event has been canceled. Other statuses should be considered informational.
(Read only property)

5.4.24 structuredLocation as EKStructuredLocationMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: The event's location with a potential geocoordinate.
Notes: (Read and Write property)

5.4.25 Constants

Events Availability

Constant	Value	Description
kAvailabilityBusy	0	The event has a busy availability setting.
kAvailabilityFree	1	The event has a free availability setting.
kAvailabilityNotSupported	-1	Availability settings are not supported by the event,Äôs calendar.
kAvailabilityTentative	2	The event has a tentative availability setting.
kAvailabilityUnavailable	3	The event has an unavailable availability setting.

Status Constants

Constant	Value	Description
kStatusCanceled	3	The event has no status.
kStatusConfirmed	1	The event has no status.
kStatusNone	0	The event has no status.
kStatusTentative	2	The event has no status.

5.5 class EKEventStoreMBS

5.5.1 class EKEventStoreMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An instance of the EKEventStore class represents the Calendar database.
Notes: It is an application‚Äôs point of contact for accessing calendar and reminder data.

You must request access to an entity type after the event store is initialized with requestAccessToEntityType for data to return.

Acceptable entity types are EKEntityMaskEvent for events and EKEntityMaskReminder for reminders.
Create a predicate, or a search query for events, with the predicateForEventsWithStartDate method.
Fetch and process events that match a given predicate with the eventsMatchingPredicate: and enumerateEventsMatchingPredicate methods.
Save and delete events from the event store with the saveEvent and removeEvent methods.
Subclass of the EKObjectMBS class.
Blog Entries

- MBS Xojo Plugins, version 24.1pr1
- MBS Xojo Plugins, version 22.0pr7
- MBS Xojo Plugins, version 18.1pr5

5.5.2 Methods

5.5.3 authorizationStatusForEntityType(entityType as Integer) as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the authorization status for the given entity type.
Notes: Available in OS X v10.9 and later.

5.5.4 Available as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.8 and newer with 64-bit.

5.5.5 calendarItemsWithExternalIdentifier(identifier as string) as EKCalendarItemMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns either the event,Äôs first occurrences or the reminders with the specified external identifier.
Notes: externalIdentifier: The calendar item,Äôs external identifier.

Returns an array of calendar items with the specified identifier.

The external identifier can be obtained from the calendarItemExternalIdentifier property. There may be more than one matching calendar item due to reasons discussed in calendarItemExternalIdentifier.

5.5.6 calendarItemWithIdentifier(identifier as string) as EKCalendarItemMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns either the event,Äôs first occurrence or the reminder with the specified identifier.
Notes: Returs the reminder or the first occurrence of an event with the specified identifier.

5.5.7 calendarsForEntityType(types as Integer) as EKCalendarMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns calendars that support a given entity type, such as reminders or events.

5.5.8 calendarWithIdentifier(identifier as string) as EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the calendar with the specified identifier.

5.5.9 cancelFetchRequest(request as EKFetchRequestMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels the request to fetch reminders.
Notes: Pass the request you got from fetchRemindersMatchingPredicate method.
The fetchRemindersMatchingPredicateCompleted event will not fire.

5.5.10 commit(byref error as NSErrorMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Commits all unsaved changes to the event store.
Notes: error: The error variable to be filled with an error object.

If the commit operation succeeded, true is returned; otherwise, false. Returns true even when there are no changes to commit.

This method allows you to save batched changes to the event store. For example, if you pass false as the commit parameter to the saveCalendar, removeCalendar, saveEvent, or removeEvent methods, the changes are not saved until this method is invoked. Likewise, if you pass true as the commit parameter to the aforementioned methods, there is no need to call this method.

5.5.11 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The default constructor.
See also:

- 5.5.12 Constructor(sources() as EKSourceMBS) 164
- 5.5.13 Constructor(types as Integer) 164

5.5.12 Constructor(sources() as EKSourceMBS)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop only.

Function: Creates an event store that contains data for the specified sources.
Notes: sources: An array of sources the event source should contain. This array may include delegate sources.

An event store that contains data for a specific collection of event sources.
See also:

- 5.5.11 Constructor 164
- 5.5.13 Constructor(types as Integer) 164

5.5.13 Constructor(types as Integer)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Initializes access to the event store with support for the given entity type.
Notes: Available in OS X v10.8 and later.
Deprecated in OS X v10.9.
See also:

- 5.5.11 Constructor 164
- 5.5.12 Constructor(sources() as EKSourceMBS) 164

5.5.14 delegateSources as EKSourceMBS()

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Deprecated: This item is deprecated and should no longer be used. You can use sources instead. **Function:** The event sources delegated to the user.
Notes: By default, delegate event sources aren,Äôt included in an event store,Äôs sources. To access events and reminders in a delegate source:

- Initialize an EKEventStore using init.
- Use requestAccessToEntityType:completion: to request access to the desired entity types.
- Get the delegate sources from the event store using delegateSources.
- After the request is granted, initialize another EKEventStoreMBS using Constructor, passing the delegate stores.

5.5.15 Destructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

5.5.16 EKErrorDomain as string

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EventKit error domain.

5.5.17 EKEventStoreChangedNotification as string

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The changed notification name.
Notes: Posted whenever changes are made to the Calendar database, including adding, removing, and changing events or reminders. Individual changes are not described. When you receive this notification, you should refetch all EKEvent and EKReminder objects you have accessed, as they are considered stale. If you are actively editing an event and do not wish to refetch it unless it is absolutely necessary to do so, you can call the refresh method on it. If the method returns true, you do not need to refetch the event.

5.5.18 enumerateEventsMatchingPredicate(predicate as NSPredicateMBS, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Finds all events that match a given predicate and calls enumerateEventsMatchingPredicateUpdate event for each event found.
Notes: predicate: The search predicate. Must be created with the predicateForEventsWithStartDate:endDate:calendars method.

Only events that have been committed are included in enumeration.
Events saved using saveEvent with the commit parameter set to NO must call commit beforehand to be included.

This method is synchronous.

5.5.19 eventsMatchingPredicate(predicate as NSPredicateMBS) as EKEventMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns all events that match a given predicate.
Notes: Returns all events that match predicate, as an array of EKEvent objects.

Only events that have been committed are included in the results. Events saved using saveEvent with the commit parameter set to false must call commit beforehand to be included.

This method is synchronous.

5.5.20 eventsMatchingPredicateAsync(predicate as NSPredicateMBS, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns all events that match a given predicate.
Notes: Calls later eventsMatchingPredicateAsyncCompleted event with all events that match predicate, as an array of EKEvent objects.

Only events that have been committed are included in the results. Events saved using saveEvent with the commit parameter set to false must call commit beforehand to be included.

This method is asynchronous.

5.5.21 eventsWithExternalIdentifier(identifier as string) as EKCalendarItemMBS()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns events with the specified external identifier.
Notes: Same as calendarItemsWithExternalIdentifier, but only returning events.

5.5.22 eventWithIdentifier(identifier as string) as EKEventMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the first occurrence of an event with a given identifier.
Notes: The event corresponding to identifier, or nil if no event is found.

5.5.23 fetchRemindersMatchingPredicate(predicate as NSPredicateMBS, tag as Variant = nil) as EKFetchRequestMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches reminders matching a given predicate asynchronously.
Notes: predicate: The search predicate.

Returns a value to be used in cancelFetchRequest to cancel the request later if desired.

Only reminders that have been committed are included in the results. Reminders saved using saveReminder with the commit parameter set to NO must call commit beforehand to be included.

This method fetches reminders asynchronously.
Calls FetchedReminders event later on success.

5.5.24 fetchRemindersMatchingPredicateSync(predicate as NSPredicateMBS) as EKReminderMBS()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches reminders matching a given predicate synchronously.
Notes: Same as fetchRemindersMatchingPredicate, except that we wait for result and return it without calling an event.

Only reminders that have been committed are included in the results. Reminders saved using saveReminder with the commit parameter set to NO must call commit beforehand to be included.

5.5.25 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Fetches completed reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate methods.

Pass nil for startDate to find all reminders completed before endDate. Similarly, pass nil for both startDate and endDate to get all complete reminders in the specified calendars.
See also:

- 5.5.26 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 168
- 5.5.27 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 169
- 5.5.28 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 170

5.5.26 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Fetches completed reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.

endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate methods.

Pass nil for startDate to find all reminders completed before endDate. Similarly, pass nil for both startDate and endDate to get all complete reminders in the specified calendars.
See also:

- 5.5.25 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date) as NSPredicateMBS 168
- 5.5.27 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 169
- 5.5.28 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 170

5.5.27 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches completed reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate methods.

Pass nil for startDate to find all reminders completed before endDate. Similarly, pass nil for both startDate and endDate to get all complete reminders in the specified calendars.
See also:

- 5.5.25 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date) as NSPredicateMBS 168
- 5.5.26 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 168
- 5.5.28 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 170

5.5.28 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches completed reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate methods.

Pass nil for startDate to find all reminders completed before endDate. Similarly, pass nil for both startDate and endDate to get all complete reminders in the specified calendars.
See also:

- 5.5.25 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date) as NSPredicateMBS 168
- 5.5.26 predicateForCompletedRemindersWithCompletionDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 168
- 5.5.27 predicateForCompletedRemindersWithCompletionDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 169

5.5.29 predicateForEvents(startDate as date, endDate as date) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a predicate for finding events in the event store that fall within a given date range.
Notes: startDate: The start date of the range of events fetched.
endDate: The end date of the range of events fetched.
calendars: Optional. The calendars to search, as an array of EKCalendarMBS objects. Passing nil indicates to search all calendars.
See also:

- 5.5.30 predicateForEvents(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 171
- 5.5.31 predicateForEvents(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 171
- 5.5.32 predicateForEvents(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 171

5.5.30 predicateForEvents(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a predicate for finding events in the event store that fall within a given date range.
Notes: startDate: The start date of the range of events fetched.
endDate: The end date of the range of events fetched.
calendars: Optional. The calendars to search, as an array of EKCalendarMBS objects. Passing nil indicates to search all calendars.
See also:

- 5.5.29 predicateForEvents(startDate as date, endDate as date) as NSPredicateMBS 170
- 5.5.31 predicateForEvents(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 171
- 5.5.32 predicateForEvents(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 171

5.5.31 predicateForEvents(startDate as dateTime, endDate as dateTime) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a predicate for finding events in the event store that fall within a given date range.
Notes: startDate: The start date of the range of events fetched.
endDate: The end date of the range of events fetched.
calendars: Optional. The calendars to search, as an array of EKCalendarMBS objects. Passing nil indicates to search all calendars.
See also:

- 5.5.29 predicateForEvents(startDate as date, endDate as date) as NSPredicateMBS 170
- 5.5.30 predicateForEvents(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 171
- 5.5.32 predicateForEvents(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 171

5.5.32 predicateForEvents(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a predicate for finding events in the event store that fall within a given date range.

Notes: startDate: The start date of the range of events fetched.
endDate: The end date of the range of events fetched.
calendars: Optional. The calendars to search, as an array of EKCalendarMBS objects. Passing nil indicates to search all calendars.
See also:

- 5.5.29 predicateForEvents(startDate as date, endDate as date) as NSPredicateMBS 170
- 5.5.30 predicateForEvents(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 171
- 5.5.31 predicateForEvents(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 171

5.5.33 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Fetches incomplete reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate:completion:.

Pass nil for startDate to find all reminders due before endDate. Similarly, pass nil for both startDate and endDate to get all incomplete reminders in the specified calendars.
See also:

- 5.5.34 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 172
- 5.5.35 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 173
- 5.5.36 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 174

5.5.34 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Fetches incomplete reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.

calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate:completion:.

Pass nil for startDate to find all reminders due before endDate. Similarly, pass nil for both startDate and endDate to get all incomplete reminders in the specified calendars.
See also:

- 5.5.33 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date) as NSPredicateMBS 172
- 5.5.35 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 173
- 5.5.36 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 174

5.5.35 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches incomplete reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate:completion:.

Pass nil for startDate to find all reminders due before endDate. Similarly, pass nil for both startDate and endDate to get all incomplete reminders in the specified calendars.
See also:

- 5.5.33 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date) as NSPredicateMBS 172
- 5.5.34 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 172
- 5.5.36 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS 174

5.5.36 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime, calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches incomplete reminders in a set of calendars within an optional range.
Notes: startDate: The starting bound of the range to search.
endDate: The ending bound of the range to search.
calendars: Optional. An array of calendars to search.

The created predicate to be used for fetchRemindersMatchingPredicate:completion:.

Pass nil for startDate to find all reminders due before endDate. Similarly, pass nil for both startDate and endDate to get all incomplete reminders in the specified calendars.
See also:

- 5.5.33 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date) as NSPredicateMBS 172
- 5.5.34 predicateForIncompleteRemindersWithDueDate(startDate as date, endDate as date, calendars() as EKCalendarMBS) as NSPredicateMBS 172
- 5.5.35 predicateForIncompleteRemindersWithDueDate(startDate as dateTime, endDate as dateTime) as NSPredicateMBS 173

5.5.37 predicateForRemindersInCalendar(calendar as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all reminders in a calendar.
Notes: The created predicate to be used for fetchRemindersMatchingPredicate methods.

5.5.38 predicateForRemindersInCalendars(calendars() as EKCalendarMBS) as NSPredicateMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all reminders in a set of calendars.
Notes: The created predicate to be used for fetchRemindersMatchingPredicate methods.

5.5.39 refreshSourcesIfNecessary

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Pulls new data from remote sources if necessary.
Notes: Use this method to pull new data from remote sources if the local data is out of date.

5.5.40 remindersWithExternalIdentifier(identifier as string) as EKCalendarItemMBS()

Plugin Version: 18.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns reminders with the specified external identifier.
Notes: Same as calendarItemsWithExternalIdentifier, but only returning reminders.

5.5.41 reminderWithIdentifier(identifier as string) as EKCalendarItemMBS

Plugin Version: 18.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the reminder with the specified identifier.
Notes: Same as calendarItemWithIdentifier, but for reminders only.

5.5.42 removeCalendar(calendar as EKCalendarMBS, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a calendar from the event store by either batching or committing the changes.
Notes: calendar: The calendar to be removed.
commit: True to remove the calendar immediately; otherwise, the change is batched until the commit: method is invoked.
error: The error that occurred, if any; otherwise, nil.

Returns ture if successful; otherwise, false.
This method raises an exception if calendar belongs to another event store.

5.5.43 removeEvent(event as EKEventMBS, span as integer, byref error as NSErrorMBS) as boolean

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: Removes an event from the event store.
Notes: event: The event to remove.
span: The span that indicates whether to remove a single event or all future instances of the event in the case of a recurring event.
error: A pointer to an error object. If an error occurs, this pointer provides access to the error. If you don,Äôt need the error information, pass nil.

Return true if the event store successfully removes the event; otherwise, NO. This method also returns false if event isn,Äôt in the event store.

This method raises an exception if the event belongs to another event store.

Same as other saveEvent, but with commit = false.
See also:

- 5.5.44 removeEvent(event as EKEventMBS, span as Integer, commit as boolean, byref error as NSErrorMBS) as boolean 176

5.5.44 removeEvent(event as EKEventMBS, span as Integer, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes an event or recurring events from the event store by either batching or committing the changes.
Notes: event: The event to remove.
span: The span to use. Indicates whether the remove affects future instances of the event in the case of a recurring event.
commit: True to remove the event immediately; otherwise, the change is batched until the commit: method is invoked.
error: The error that occurred, if any did. Otherwise, nil.

If the event has successfully removed, true; otherwise, false. Also returns false if event cannot be removed because it is not in the event store.

This method raises an exception if it is passed an event from another event store.
See also:

- 5.5.43 removeEvent(event as EKEventMBS, span as integer, byref error as NSErrorMBS) as boolean 175

5.5.45 removeReminder(reminder as EKReminderMBS, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a reminder from the event store by either committing or batching the changes.
Notes: reminder: The reminder to be removed.
commit: A Boolean value indicating whether to remove the reminder immediately or to batch the removals; passing NO will not commit the removal from the event store until the commit: method is invoked.
error: The error that occurred, if any; otherwise, nil.

If successful, true; otherwise, false.
This method raises an exception if reminder belongs to another event store.

5.5.46 requestAccessToEntityType(entityType as Integer, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Prompts the user to grant or deny access to event or reminder data.
Notes: entityType: The event or reminder entity type.

Requesting access to an event store asynchronously prompts your users for permission to use their data. The user is only prompted the first time your app requests access to an entity type; any subsequent instantiations of EKEventStore uses existing permissions. When the user taps to grant or deny access, the requestAccessToEntityTypeCompleted event will be called on an arbitrary queue. Your app is not blocked while the user decides to grant or deny permission.

After users choose their permission level, the event store either calls the event.
Available in OS X v10.9 and later.

5.5.47 requestFullAccessToEvents(tag as variant = nil)

Plugin Version: 24.1, Platform: macOS, Targets: Desktop & iOS.

Function: Prompts people to grant or deny read and write access to event data.
Notes: Requesting access to an event store asynchronously prompts people for permission to use their data. The operating system only prompts them the first time your app requests full access to events; any subsequent instantiations of EKEventStore uses existing permissions. When they grant or deny access, EventKit calls the requestAccessToEntityTypeCompleted event.

Your app isn,Äôt blocked while the person decides to grant or deny permission. Because they may deny permission, your app should handle cases where it doesn,Äôt receive access to the event store.

5.5.48 requestFullAccessToReminders(tag as variant = nil)

Plugin Version: 24.1, Platform: macOS, Targets: Desktop & iOS.

Function: Prompts people to grant or deny read and write access to reminders.
Notes: Requesting access to an event store asynchronously prompts people for permission to use their data. The operating system only prompts them the first time your app requests access to reminders; any subsequent instantiations of EKEventStore uses existing permissions. When they grant or deny access, EventKit calls the requestAccessToEntityTypeCompleted event.

Your app isn‚Äôt blocked while the person decides to grant or deny permission. Because they may deny permission, your app should handle cases where it doesn‚Äôt receive access to the event store.

5.5.49 requestWriteOnlyAccessToEvents(tag as variant = nil)

Plugin Version: 24.1, Platform: macOS, Targets: Desktop & iOS.

Function: Prompts the person using your app to grant or deny write access to event data.
Notes: Requesting access to an event store asynchronously prompts people for permission to use their data. The operating system only prompts them the first time your app requests write-only event access; any subsequent instantiations of EKEventStore uses existing permissions. When they grant or deny access, EventKit calls the requestAccessToEntityTypeCompleted event.

Your app isn‚Äôt blocked while the person decides to grant or deny permission. Because they may deny permission, your app should handle cases where it doesn‚Äôt receive access to the event store.

5.5.50 reset

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the event store to its saved state.
Notes: This method updates all the properties of all the objects with their corresponding values in the event store. Any local changes that were not saved before invoking this method will be lost. All objects that were created or retrieved using this store are disassociated from it and should be considered invalid.

5.5.51 saveCalendar(calendar as EKCalendarMBS, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Saves a calendar to the event store by either committing or batching the changes.
Notes: calendar: The calendar to be saved.
commit: True to save the calendar immediately; otherwise, the change is batched until the commit method is invoked.
error: The error that occurred, if any; otherwise, nil.

Returns true if successful; otherwise, false.

This method raises an exception if calendar belongs to another event store.

5.5.52 saveEvent(event as EKEventMBS, span as integer, byref error as NSErrorMBS) as boolean

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: Saves changes to an event permanently.
Notes: event: The event to save.
span: The span that indicates whether to remove a single event or all future instances of the event in the case of a recurring event.
error: A pointer to an error object. If an error occurs, this pointer provides access to the error. If you don,Äôt need the error information, pass nil.

Returns true if the event store saves the event successfully; otherwise, NO. This method also returns false if event wasn,Äôt modified.

This method raises an exception if it,Äôs passed an event from another event store.
When an event is saved, it,Äôs updated in the Calendar database. Any fields you didn,Äôt modify are updated to reflect the most recent value in the database. If the event has been deleted from the database, it,Äôs recreated as a new event.

Same as other saveEvent, but with commit = false.
See also:

- 5.5.53 saveEvent(event as EKEventMBS, span as Integer, commit as boolean, byref error as NSErrorMBS) as boolean 179

5.5.53 saveEvent(event as EKEventMBS, span as Integer, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Saves an event or recurring events to the event store by either batching or committing the

changes.
Notes: event: The event to be saved.
span: The span to use. Indicates whether the save affects future instances of the event in the case of a recurring event.
commit: To save the event immediately, pass true; otherwise, the change is batched until the commit: method is invoked.
error: The error that occurred, if any; otherwise, nil.

Returns if successful, true; otherwise, false. Also returns false if event does not need to be saved because it has not been modified.

This method raises an exception if it is passed an event from another event store.

When an event is saved, it is updated in the Calendar database. Any fields you did not modify are updated to reflect the most recent value in the database. If the event has been deleted from the database, it is re-created as a new event.
See also:

- 5.5.52 saveEvent(event as EKEventMBS, span as integer, byref error as NSErrorMBS) as boolean 179

5.5.54 saveReminder(reminder as EKReminderMBS, commit as boolean, byref error as NSErrorMBS) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Saves changes to a reminder by either committing or batching the changes.
Notes: reminder: The reminder to be saved.
commit: A Boolean value indicating whether to save the reminder immediately or to batch the changes; passing NO will not commit changes to the event store until the commit: method is invoked.
error: The error that occurred, if any; otherwise, nil.

If successful, true; otherwise, false.
This method raises an exception if reminder belongs to another event store.

5.5.55 sources as EKSourceMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an unordered array of source objects.
Notes: An EKSource object represents an account that contains calendars.

5.5.56 sourceWithIdentifier(identifier as string) as EKSourceMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a source with the specified identifier.

5.5.57 Properties

5.5.58 defaultCalendarForNewEvents as EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar that events are added to by default, as specified by user settings.
Notes: (Read only property)

5.5.59 defaultCalendarForNewReminders as EKCalendarMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The calendar that events are added to by default, as specified by user settings.
Notes: (Read only property)

5.5.60 eventStoreIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The unique identifier for the event store.
Notes: If the store is damaged, it is re-created and given a new identifier. If this value is different from a fetched value, you should take the appropriate action.
(Read only property)

5.5.61 Events

5.5.62 Changed

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The changed event.
Notes: Posted whenever changes are made to the Calendar database, including adding, removing, and changing events or reminders. Individual changes are not described. When you receive this notification, you should refetch all EKEvent and EKReminder objects you have accessed, as they are considered stale. If you

are actively editing an event and do not wish to refetch it unless it is absolutely necessary to do so, you can call the refresh method on it. If the method returns true, you do not need to refetch the event.

5.5.63 enumerateEventsMatchingPredicateUpdate(e as EKEventMBS, byref stop as boolean, predicate as NSPredicateMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The method enumerateEventsMatchingPredicate found another event.

5.5.64 eventsMatchingPredicateAsyncCompleted(events() as EKEventMBS, predicate as NSPredicateMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The async call to eventsMatchingPredicate completed.

5.5.65 fetchRemindersMatchingPredicateCompleted(reminders() as EKReminderMBS, predicate as NSPredicateMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The call to fetchRemindersMatchingPredicate completed.

5.5.66 requestAccessToEntityTypeCompleted(entityType as Integer, granted as Boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The event called by requestAccessToEntityType once decision is made.

5.5.67 Constants

Authorization Status

Constant	Value	Description
kAuthorizationStatusAuthorized	3	The app is authorized to access the service.
kAuthorizationStatusDenied	2	The user explicitly denied access to the service for the app.
kAuthorizationStatusNotDetermined	0	The user has not yet made a choice regarding whether the app may access service.
kAuthorizationStatusRestricted	1	The app is not authorized to access the service. The user cannot cha this app,Äôs authorization status, possibly due to active restrictions suc parental controls being in place.

Entity Mask

Constant	Value	Description
kEntityMaskEvent	1	Event
kEntityMaskReminder	2	Reminder

Types

Constant	Value	Description
kEntityTypeEvent	0	Entity Event
kEntityTypeReminder	1	Entity Reminder

Errors

Constant	Value	Description
kErrorAlarmGreaterThanRecurrence	8	The alarm interval is greater than the recurrence interval.
kErrorAlarmProximityNotSupported	21	The source does not allow geofences on alarms.
kErrorCalendarDoesNotAllowEvents	22	The calendar does not allow events to be added.
kErrorCalendarDoesNotAllowReminders	23	The calendar does not allow reminders to be added.
kErrorCalendarHasNoSource	14	The calendar cannot be saved with no source set.
kErrorCalendarIsImmutable	16	The calendar is immutable and cannot be modified or deleted.
kErrorCalendarReadOnly	6	The calendar is read-only and cannot have events added to it.
kErrorCalendarSourceCannotBeModified	15	The calendar cannot be moved to another source.
kErrorDatesInverted	4	The event,Äôs end date occurs before its start date.
kErrorDurationGreaterThanRecurrence	7	The duration of an event is greater than its recurrence interval.
kErrorEventNotMutable	0	The event is not mutable and cannot be saved or deleted.
kErrorInternalFailure	5	An internal error occurred.
kErrorInvalidSpan	13	An invalid span was passed when saving or deleting.
kErrorInvitesCannotBeMoved	12	The event cannot be moved because it is an invite.
kErrorNoCalendar	1	The event is not associated with a calendar.
kErrorNoEndDate	3	The event has no end date set.
kErrorNoStartDate	2	The event has no start date set.
kErrorObjectBelongsToDifferentStore	11	The object belongs to a different calendar store.
kErrorRecurringReminderRequiresDueDate	18	The recurring reminder requires a due date.
kErrorReminderLocationsNotSupported	20	The source doesn't support locations on reminders.
kErrorSourceDoesNotAllowCalendarAddDelete	17	The source doesn,Äôt allow calendars to be added or deleted.
kErrorSourceDoesNotAllowEvents	25	The source does not allow calendars supporting event entity typ
kErrorSourceDoesNotAllowReminders	24	The source does not allow calendars supporting reminder entity
kErrorStartDateCollidesWithOtherOccurrence	10	The event,Äôs start date collides with another occurrence of the
kErrorStartDateTooFarInFuture	9	The start date is further into the future than the calendar can c
kErrorStructuredLocationsNotSupported	19	The source to which this calendar belongs does not support st tions.

Event Span

Constant	Value	Description
kSpanFutureEvents	1	Modifications to this event instance should also affect future instances of this event.
kSpanThisEvent	0	Modifications to this event instance should affect only this instance.

Week Days

Constant	Value	Description
kWeekDayFriday	6	Friday
kWeekDayMonday	2	Monday
kWeekDaySaturday	7	Saturday
kWeekDaySunday	1	Sunday
kWeekDayThursday	5	Thursday
kWeekDayTuesday	3	Tuesday
kWeekDayWednesday	4	Wednesday

5.6 class EKFetchRequestMBS

5.6.1 class EKFetchRequestMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The class for an ongoing fetch report.
Notes: If request object is destroyed, the request is cancelled, if it is not yet finished.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.6.2 Methods

5.6.3 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

5.6.4 Properties

5.6.5 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

5.7 class EKObjectMBS

5.7.1 class EKObjectMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EKObject class is an abstract superclass for all Event Kit classes whose instances are persistent.
Notes: It provides fine control when saving and restoring property settings. For example, you can find out if a persistent object was modified locally and whether it needs to be saved. If the object has changed in the event store since it was fetched, you can refresh the local copy by keeping local changes or by removing local changes. You can also roll back the object to the state when it was first fetched.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.7.2 Methods

5.7.3 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

5.7.4 hasChanges as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns whether this object or any of the objects it contains has uncommitted changes.
Notes: Returns true if there are uncommitted changes; otherwise, false.

5.7.5 isNew as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns whether this object has ever been saved.
Notes: Returns true if the object has not been saved; otherwise, false.

5.7.6 refresh as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Merges changes to this object with the latest saved values.
Notes: If the operation is successful, YES; if the object was deleted in the event store, false. If this method returns false, the object should be released.

This method merges the local changes to properties of this object with the latest values in the event store. This method updates only properties that have not been modified locally, so you do not lose any changes by invoking this method. You can also use this method to see whether an object was deleted from the event store.

5.7.7 reset

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns this object to its saved state.
Notes: This method updates all the properties of this object with the corresponding values in the event store. Any local changes that were not saved before invoking this method are lost. This method does nothing if the object was never saved.

5.7.8 rollback

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Rolls back the property values of this object to its original state when it was first fetched.
Notes: Any local changes to this object are lost when invoking this method. This method does not refetch property values from the event store. This method does nothing if the object was never changed.

5.7.9 Properties

5.7.10 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

5.8 class EKParticipantMBS

5.8.1 class EKParticipantMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An EKParticipant object represents a participant in a calendar event.
Notes: A participant can be a person, group, room, or other resource.

Do not create EKParticipant objects directly. Instead, use the property attendees on EKCalendarItem to return an array of EKParticipant objects.

Event Kit cannot add participants to an event nor change participant information. Use the properties in this class to get information about a participant.
Subclass of the EKObjectMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 22.0pr7

5.8.2 Methods

5.8.3 ABPersonInAddressBook(addressBook as Variant) as Variant

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Returns the address book record that represents the participant.
Notes: Please pass a ABAddressbookMBS object for the addressBook parameter. You get back a ABPersonMBS object.
The address book record for the participant, or nil if the record is not found.
This method searches for a record match based on the participant,Äôs email address.

5.8.4 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

5.8.5 copy as EKParticipantMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the participant.

5.8.6 Properties

5.8.7 contactPredicate as NSPredicateMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: A predicate to use with the Contacts framework to retrieve the corresponding contact instance.
Notes: Use this property to get a predicate that you can use with a CNContactStoreMBS to fetch a CNContactMBS instance for this participant, if one exists.
(Read only property)

5.8.8 isCurrentUser as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether this participant represents the owner of this account.
Notes: (Read only property)

5.8.9 name as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The participant,Äôs name.
Notes: (Read only property)

5.8.10 participantRole as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The participant,Äôs role in the event.
Notes: (Read only property)

5.8.11 participantStatus as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The participant,Äôs attendance status.
Notes: (Read only property)

5.8.12 participantType as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The participant,Äôs type.
Notes: (Read only property)

5.8.13 person as Variant

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The person in the shared addressbook instance.
Notes: Value is an ABPersonMBS object.
(Read only property)

5.8.14 URL as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The URL representing this participant.
Notes: (Read only property)

5.8.15 Constants

Roles

Constant	Value	Description
kRoleChair	3	The participant is the chair of the event.
kRoleNonParticipant	4	The participant does not have an active role in the event.
kRoleOptional	2	The participant,Äôs attendance is optional.
kRoleRequired	1	The participant,Äôs attendance is required.
kRoleUnknown	0	The participant,Äôs role is unknown.

Status Constants

Constant	Value	Description
kStatusAccepted	2	The participant has accepted the event.
kStatusCompleted	6	The participant,Äôs event has completed.
kStatusDeclined	3	The participant has declined the event.
kStatusDelegated	5	The participant has delegated attendance to another participant.
kStatusInProcess	7	The participant,Äôs event is currently in process.
kStatusPending	1	The participant has yet to respond to the event.
kStatusTentative	4	The participant,Äôs attendance status is tentative.
kStatusUnknown	0	The participant,Äôs attendance status is unknown.

Participant Types

Constant	Value	Description
kTypeGroup	4	The participant is a group.
kTypePerson	1	The participant is a person.
kTypeResource	3	The participant is a resource.
kTypeRoom	2	The participant is a room.
kTypeUnknown	0	The participant,Äôs type is unknown.

5.9 class EKRecurrenceDayOfWeekMBS

5.9.1 class EKRecurrenceDayOfWeekMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EKRecurrenceDayOfWeek class represents a day of the week for use with an EKRecurrenceRule object.
Notes: A day of the week can optionally have a week number, indicating a specific day in the recurrence rule,Äôs frequency. For example, a day of the week with a day value of Tuesday and a week number of 2 would represent the second Tuesday of every month in a monthly recurrence rule, and the second Tuesday of every year in a yearly recurrence rule. A day of the week with a week number of 0 ignores its week number.
Blog Entries

- MBS Xojo Plugins, version 20.6pr1

5.9.2 Methods

5.9.3 Constructor(dayOfTheWeek as Integer)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a day of the week with a given day and week number.
Notes: dayOfTheWeek: The day of the week. Values range from 1 to 7, with Sunday being 1.
weekNumber: Optional, the week number.
See also:

- 5.9.4 Constructor(dayOfTheWeek as Integer, weekNumber as Integer) 192

5.9.4 Constructor(dayOfTheWeek as Integer, weekNumber as Integer)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a day of the week with a given day and week number.
Notes: dayOfTheWeek: The day of the week. Values range from 1 to 7, with Sunday being 1.
weekNumber: Optional, the week number.
See also:

- 5.9.3 Constructor(dayOfTheWeek as Integer) 192

5.9.5 copy as EKRecurrenceDayOfWeekMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of this object.

5.9.6 dayOfWeek(dayOfTheWeek as Integer) as EKRecurrenceDayOfWeekMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a day of the week with a given day.
Notes: The week number of the returned day of the week is 0.
Values range from 1 to 7, with Sunday being 1.
See also:

- 5.9.7 dayOfWeek(dayOfTheWeek as Integer, weekNumber as Integer) as EKRecurrenceDayOfWeekMBS 193

5.9.7 dayOfWeek(dayOfTheWeek as Integer, weekNumber as Integer) as EKRecurrenceDayOfWeekMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns an autoreleased day of the week with a given day and week number.
Notes: dayOfTheWeek: The day of the week. Values range from 1 to 7, with Sunday being 1.
weekNumber: The week number.
See also:

- 5.9.6 dayOfWeek(dayOfTheWeek as Integer) as EKRecurrenceDayOfWeekMBS 193

5.9.8 Properties

5.9.9 dayOfTheWeek as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The day of the week.
Notes: Values are from 1 to 7, with Sunday being 1.
(Read only property)

5.9.10 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

5.9.11 weekNumber as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The week number of the day of the week.
Notes: Values range from –53 to 53. A negative value indicates a value from the end of the range. 0 indicates the week number is irrelevant.
(Read only property)

5.9.12 Constants

Days

Constant	Value	Description
kFriday	6	Friday
kMonday	2	Monday
kSaturday	7	Saturday
kSunday	1	Sunday
kThursday	5	Thursday
kTuesday	3	Tuesday
kWednesday	4	Wednesday

5.10 class EKRecurrenceEndMBS

5.10.1 class EKRecurrenceEndMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EKRecurrenceEnd class defines the end of a recurrence rule defined by an EKRecurrenceRule object.
Notes: The recurrence end can be specified by a date (date-based) or by a maximum count of occurrences (count-based). An event which is set to never end should have its EKRecurrenceEnd set to nil.

5.10.2 Methods

5.10.3 Constructor(endDate as date)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Initializes and returns a date-based recurrence end with a given end date.
Notes: The end date argument must be a valid NSDate and not nil; otherwise an exception will be raised.
See also:

- 5.10.4 Constructor(endDate as dateTime) 195
- 5.10.5 Constructor(occurrenceCount as Integer) 195

5.10.4 Constructor(endDate as dateTime)

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a date-based recurrence end with a given end date.
Notes: The end date argument must be a valid NSDate and not nil; otherwise an exception will be raised.
See also:

- 5.10.3 Constructor(endDate as date) 195
- 5.10.5 Constructor(occurrenceCount as Integer) 195

5.10.5 Constructor(occurrenceCount as Integer)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a count-based recurrence end with a given maximum occurrence count.
Notes: The maximum occurrence count argument must be a positive integer and not 0; otherwise an exception will be raised.
See also:

5.10.6 copy as EKRecurrenceEndMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

5.10.7 recurrenceEndWithEndDate(endDate as date) as EKRecurrenceEndMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Initializes and returns a date-based recurrence end with a given end date.
Notes: The end date argument must be a valid NSDate and not nil; otherwise an exception will be raised.

5.10.8 recurrenceEndWithOccurrenceCount(occurrenceCount as Integer) as EKRecurrenceEndMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Initializes and returns a count-based recurrence end with a given maximum occurrence count.
Notes: The maximum occurrence count argument must be a positive integer and not 0; otherwise an exception will be raised.

5.10.9 Properties

5.10.10 endDate as Date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The end date of the recurrence end, or nil if the recurrence end is count-based.
Notes: (Read only property)

5.10.11 endDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The end date of the recurrence end, or nil if the recurrence end is count-based.
Notes: (Read only property)

5.10.12 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

5.10.13 occurrenceCount as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The occurrence count of the recurrence end, or 0 if the recurrence end is date-based.
Notes: (Read only property)

5.11 class EKRecurrenceRuleMBS

5.11.1 class EKRecurrenceRuleMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EKRecurrenceRule class is used to describe the recurrence pattern for a recurring event.
Notes: After you create a recurrence rule, assign it to an event with the recurrenceRule method of EKEvent.

Recurrence rules can have an end, represented by an EKRecurrenceEnd object. The end can be based on a specific date or a maximum number of occurrences.
Subclass of the EKObjectMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.6pr1

5.11.2 Methods

5.11.3 Constructor(type as Integer, interval as Integer, days() as EKRecurrenceDayOfWeekMBS, monthDays() as Integer, months() as Integer, weeksOfTheYear() as Integer, daysOfTheYear() as Integer, setPositions() as Integer, end as EKRecurrenceEndMBS = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a recurrence rule with a given frequency and additional scheduling information.
Notes: type: The frequency of the recurrence rule. Can be daily, weekly, monthly, or yearly.
interval: The interval between instances of this recurrence. For example, a weekly recurrence rule with an interval of 2 occurs every other week. Must be greater than 0.
days: The days of the week that the event occurs, as an array of EKRecurrenceDayOfWeek objects.
monthDays: The days of the month that the event occurs, as an array of integers. Values can be from 1 to 31 and from -1 to -31. This parameter is only valid for recurrence rules of type EKRecurrenceFrequencyMonthly.
months: the months of the year that the event occurs, as an array of integers. Values can be from 1 to 12. This parameter is only valid for recurrence rules of type EKRecurrenceFrequencyYearly.
weeksOfTheYear: The weeks of the year that the event occurs, as an array of integers. Values can be from 1 to 53 and from -1 to -53. This parameter is only valid for recurrence rules of type EKRecurrenceFrequencyYearly.
daysOfTheYear: The days of the year that the event occurs, as an array of integers. Values can be from 1 to 366 and from -1 to -366. This parameter is only valid for recurrence rules of type EKRecurrenceFrequencyYearly.
setPositions: An array of ordinal numbers that filters which recurrences to include in the recurrence rule,Äôs frequency. See setPositions for more information.
end: The end of the recurrence rule.

Handle is zero if invalid values are provided.

Negative values indicate counting backwards from the end of the recurrence rule,Äôs frequency.
See also:

- 5.11.4 Constructor(type as Integer, interval as Integer, end as EKRecurrenceEndMBS = nil) 199

5.11.4 Constructor(type as Integer, interval as Integer, end as EKRecurrenceEndMBS = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a simple recurrence rule with a given frequency, interval, and end.
Notes: type: The frequency of the recurrence rule. Can be daily, weekly, monthly, or yearly.
interval: The interval between instances of this recurrence. For example, a weekly recurrence rule with an interval of 2 occurs every other week. Must be greater than 0.
end: The end of the recurrence rule.

Handle is zero if invalid values are provided.
See also:

- 5.11.3 Constructor(type as Integer, interval as Integer, days() as EKRecurrenceDayOfWeekMBS, monthDays() as Integer, months() as Integer, weeksOfTheYear() as Integer, daysOfTheYear() as Integer, setPositions() as Integer, end as EKRecurrenceEndMBS = nil) 198

5.11.5 copy as EKRecurrenceRuleMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the recurrence rule.

5.11.6 daysOfTheMonth as Integer()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The days of the month associated with the recurrence rule, as an array of integers.
Notes: Values can be from 1 to 31 and from -1 to -31.This property value is valid only for recurrence rules that were initialized with specific days of the month and a frequency type of EKRecurrenceFrequency-Monthly.
Negative values indicate counting backwards from the end of the month.

5.11.7 daysOfTheWeek as EKRecurrenceDayOfWeekMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The days of the week associated with the recurrence rule, as an array of EKRecurrenceDayOfWeek objects.
Notes: This property value is valid only for recurrence rules that were initialized with specific days of the week and a frequency type of EKRecurrenceFrequencyWeekly, EKRecurrenceFrequencyMonthly, or EKRecurrenceFrequencyYearly.

5.11.8 daysOfTheYear as Integer()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The days of the year associated with the recurrence rule, as an array of integers.
Notes: Values can be from 1 to 366 and from -1 to -366. This property value is valid only for recurrence rules initialized with a frequency type of EKRecurrenceFrequencyYearly.

Negative values indicate counting backwards from the end of the year.

5.11.9 monthsOfTheYear as Integer()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The months of the year associated with the recurrence rule, as an array of integers.
Notes: Values can be from 1 to 12. This property value is valid only for recurrence rules initialized with specific months of the year and a frequency type of EKRecurrenceFrequencyYearly.

5.11.10 setPositions as Integer()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of ordinal numbers that filters which recurrences to include in the recurrence rule,Äôs frequency.
Notes: For example, a yearly recurrence rule that has a daysOfTheWeek value that specifies Monday through Friday, and a setPositions array containing 2 and -1, occurs only on the second weekday and last weekday of every year.

Values can be from 1 to 366 and from -1 to -366.

Negative values indicate counting backwards from the end of the recurrence rule,Äôs frequency (week, month,

or year).

5.11.11 weeksOfTheYear as Integer()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The weeks of the year associated with the recurrence rule, as an array of integers.
Notes: Values can be from 1 to 53 and from -1 to -53. This property value is valid only for recurrence rules initialized with specific weeks of the year and a frequency type of EKRecurrenceFrequencyYearly.
Negative values indicate counting backwards from the end of the year.

5.11.12 Properties

5.11.13 calendarIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The identifier for the recurrence rule,Äôs calendar.
Notes: (Read only property)

5.11.14 firstDayOfTheWeek as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates which day of the week the recurrence rule treats as the first day of the week.
Notes: Values of 1 to 7 correspond to Sunday through Saturday. A value of 0 indicates that this property is not set for the recurrence rule.
(Read only property)

5.11.15 frequency as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The frequency of the recurrence rule.
Notes: (Read only property)

5.11.16 interval as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies how often the recurrence rule repeats over the unit of time indicated by its frequency.
Notes: For example, a recurrence rule with a frequency type of EKRecurrenceFrequencyWeekly and an interval of 2 repeats every two weeks.
(Read only property)

5.11.17 recurrenceEnd as EKRecurrenceEndMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates when the recurrence rule ends.
Notes: This can be represented by an end date or a number of occurrences.
(Read and Write property)

5.11.18 Constants

Frequencies for recurrence rules.

Constant	Value	Description
kRecurrenceFrequencyDaily	0	Indicates a daily recurrence rule.
kRecurrenceFrequencyMonthly	2	Indicates a monthly recurrence rule.
kRecurrenceFrequencyWeekly	1	Indicates a weekly recurrence rule.
kRecurrenceFrequencyYearly	3	Indicates a yearly recurrence rule.

5.12 class EKReminderMBS

5.12.1 class EKReminderMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An instance of the EKReminder class represents a reminder added to a calendar in the Event Kit framework.
Notes: Use the reminderWithEventStore: method to create a new reminder. Use the properties in the class to get and modify certain information about a reminder.
Subclass of the EKCalendarItemMBS class.

5.12.2 Methods

5.12.3 Constructor(eventStore as EKEventStoreMBS)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a new reminder in the given event store.

5.12.4 reminderWithEventStore(eventStore as EKEventStoreMBS) as EKReminderMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates and returns a new reminder in the given event store.

5.12.5 Properties

5.12.6 completed as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value determining whether or not the reminder is marked completed.
Notes: Setting this property to true will set completionDate to the current date; setting this property to false will set completionDate to nil.

If the reminder was completed using a different client, you may encounter the case where this property is true, but completionDate is nil.
(Read and Write property)

5.12.7 completionDate as date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The completion date as a date.
Notes: (Read and Write property)

5.12.8 completionDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The completion date as a date.
Notes: (Read and Write property)

5.12.9 dueDate as date

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The due date.
Notes: (Read only property)

5.12.10 dueDateComponents as NSDateComponentsMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The date by which the reminder should be completed.
Notes: The use of date components allows the due date and its time zone to be represented in a single property. A nil time zone represents a floating date. Setting a date component without an hour, minute and second component will set the reminder to be an all-day reminder. If this property is set, the calendar must be set to NSGregorianCalendar; otherwise an exception is raised.

This components,Äôs timeZone property is independent of time zone properties on startDateComponents and its super EKCalendarItem object. By default, the due date is set to the system time zone.

On iOS, Event Kit requires that a start date is set if the due date is set, however this is not a requirement on OS X.
(Read and Write property)

5.12.11 dueDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The due date.
Notes: (Read only property)

5.12.12 priority as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The priority of the reminder.
Notes: Priorities run from 1 (highest) to 9 (lowest). A priority of 0 means no priority.
Saving a reminder with any other priority will fail.
Per RFC 5545, priorities of 1-4 are considered ”high,” a priority of 5 is ”medium,” and priorities of 6-9 are ”low.”
(Read and Write property)

5.12.13 startDateComponents as NSDateComponentsMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The start date of the task.
Notes: The use of date components allows the start date and its time zone to be represented in a single property. A nil time zone represents a floating date. Setting a date component without an hour, minute and second component will set the reminder to be an all-day reminder. If this property is set, the calendar must be set to NSGregorianCalendar; otherwise an exception is raised.

The start date components,Äôs timeZone property corresponds to the timeZone property on EKCalendarItem. A change in one value will cause a change in the other. Setting the time zone directly on the components does not guarantee that your changes will be saved; instead, pull this property from the reminder, set the time zone on it, and assign it back to the reminder:
(Read and Write property)

5.12.14 Constants

Priorities

Constant	Value	Description
kPriorityHigh	1	High
kPriorityLow	9	Low
kPriorityMedium	5	Medium
kPriorityNone	0	No priority

5.13 class EKSourceMBS

5.13.1 class EKSourceMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: An instance of the EKSource class represents the account that a calendar belongs to.
Notes: You do not create instances of this class. You retrieve EKSource objects from an EKEventStore object. Use the sources property to get all the EKSource objects for an event store, and use the methods in this class to access properties of the source object.
Subclass of the EKObjectMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.13.2 Methods

5.13.3 calendarsForEntityType(types as Integer) as EKCalendarMBS()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the calendars that belong to this source object that support a particular entity type.
Notes: entityType: The entity type of either an event or a reminder.

5.13.4 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

5.13.5 Properties

5.13.6 sourceIdentifier as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A unique identifier for the source object.
Notes: (Read only property)

5.13.7 sourceType as Integer

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The type of this source object.
Notes: (Read only property)

5.13.8 title as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of this source object.
Notes: (Read only property)

5.13.9 Constants

Types

Constant	Value	Description
kTypeBirthdays	5	Represents a birthday source.
kTypeCalDAV	2	Represents a CalDAV or iCloud source.
kTypeExchange	1	Represents an Exchange source.
kTypeLocal	0	Represents a local source.
kTypeMobileMe	3	Represents a MobileMe source.
kTypeSubscribed	4	Represents a subscribed source.

5.14 class EKStructuredLocationMBS

5.14.1 class EKStructuredLocationMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The EKStructuredLocation class specifies a geofence that can be used to trigger the alarm of a calendar item.
Notes: Use locationWithTitle to create a new structured location, then set it to the structuredLocation property of an EKAlarm object.
Subclass of the EKObjectMBS class.
Blog Entries

- MBS Xojo Plugins, version 22.0pr7

Xojo Developer Magazine

- 21.1, page 29: News from MBS Xojo Plugins, What's up with MonkeyBread Software by Stefanie Juchmes

5.14.2 Methods

5.14.3 Constructor(title as string)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new structured location.
Notes: title: The title of the location.

5.14.4 copy as EKObjectMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the location.

5.14.5 locationWithMapItem(MapItem as Variant) as EKStructuredLocationMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop only.

Function: Creates a new structured location with the specified map item.
Notes: mapItem: The MKMapItemMBS object that represents the location.

5.14.6 locationWithTitle(title as string) as EKStructuredLocationMBS

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new structured location.
Notes: title: The title of the location.

5.14.7 Properties

5.14.8 geoLocation as Variant

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The core location.
Notes: Value is a CLLocationMBS object.
(Read and Write property)

5.14.9 radius as Double

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: A minimum distance from the core location that would trigger the alarm or reminder.
Notes: To use the default radius, set this property to 0.
(Read and Write property)

5.14.10 title as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop & iOS.

Function: The title of the location.
Notes: (Read and Write property)

Chapter 6

CloudKit

6.1 class CKAcceptSharesOperationMBS

6.1.1 class CKAcceptSharesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that accepts shared records.
Notes: Subclass of the CKOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 21.3pr1

6.1.2 Methods

6.1.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to process share metadata.
Notes: You can use the returned CKAcceptSharesOperation only once. When executed, this query object processes the share metadata.
See also:

- 6.1.4 Constructor(shareMetadatas() as CKShareMetadataMBS) 211

6.1.4 Constructor(shareMetadatas() as CKShareMetadataMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to process the specified share objects.
Notes: shareMetadatas: An array of CKShareMetadataMBS objects. This parameter is used to initialize the value in the shareMetadatas property. If you specify nil, you must assign an appropriate value to the shareMetadatas property before executing the operation.
See also:

- 6.1.3 Constructor 211

6.1.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.1.6 setShareMetadatas(shareMetadatas() as CKShareMetadataMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the metadata of the shares that you want to process.
Notes: Use this property to view or change the metadata of the share objects you want to process. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.1.7 shareMetadatas as CKShareMetadataMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The metadata of the shares that you want to process.
Notes: Use this property to view or change the metadata of the share objects you want to process. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.1.8 Events

6.1.9 acceptSharesCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: This event is called when the operation completes.
Notes: The Completed event of the underlying NSOperation is also called if both are set. If the error is CKErrorPartialFailure, the error,Äôs userInfo dictionary contains a dictionary of shareURL objects to errors keyed off of CKPartialErrorsByItemIDKey. This call happens as soon as the server has seen all record

changes and may be invoked while the server is processing the side effects of those changes.

The event is executed serially with respect to the other progress blocks of the operation. If you intend to use this event to process results, update the value of this property before executing the operation or submitting the operation object to a queue.

6.1.10 perShareCompleted(shareMetadata as CKShareMetadataMBS, acceptedShare as CKShareMBS, error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each share metadata that the server has processed.
Notes: If the error is nil the share was successfully accepted. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.2 class CKAssetMBS

6.2.1 class CKAssetMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The class for a file.

6.2.2 Methods

6.2.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.2.4 Constructor(file as FolderItem)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initialize an asset to be saved with the content at the given file.
See also:

- 6.2.5 Constructor(URL as String) 214

6.2.5 Constructor(URL as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initialize an asset to be saved with the content at the given file URL.
See also:

- 6.2.4 Constructor(file as FolderItem) 214

6.2.6 Properties

6.2.7 fileURL as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The URL to the file.
Notes: Local file URL where fetched records are cached and saved records originate from.
(Read only property)

6.2.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.3 class CKContainerMBS

6.3.1 class CKContainerMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An encapsulation of content associated with an app, including content that is accessible to all users and content that is available only to a specific user.
Notes: A container object manages all explicit and implicit attempts to access the contents of the container.

Every app has a default container object that manages its own native content. If you develop a suite of apps, you can also access any container objects for which you have the appropriate entitlements. Each new container distinguishes between publicly available data and data that is private to the current user. Private data is always stored in the appropriate container directory in the user,Äôs iCloud account.

see
https://developer.apple.com/reference/cloudkit/ckcontainer
Blog Entries

- News from the MBS Xojo Plugins Version 21.3
- MBS Xojo Plugins, version 21.3pr1
- MBS Xojo Plugins, version 21.2pr4

6.3.2 Methods

6.3.3 accountStatus(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Reports whether the current user,Äôs iCloud account can be accessed.
Notes: This method determines the status of the current user,Äôs iCloud account asynchronously, reporting the results to the block in the completionHandler parameter. Call this method before accessing the private database to determine whether that database is available. While your app is running, use the NSUbiquityIdentityDidChangeNotification notification to detect account changes and call this method again to retrieve the status for the new account.

6.3.4 addOperation(operation as CKOperationMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Queues an operation for execution in the current container.
Notes: operation: The CKOperationMBS to enqueue. Make sure the operation object you provide is fully

configured and ready to be enqueued. Do not change the settings of this object after calling this method.

This method adds the operation object to an operation queue managed by the container itself. Operation objects in the container,Äôs queue execute concurrently with default priorities in background threads.

This method changes the operation object,Äôs container (if any) to the current container.

6.3.5 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.3.6 CKAccountChangedNotification as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The account changed notification name.

6.3.7 CKCurrentUserDefaultName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The default name for the current user.

6.3.8 CKErrorDomain as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The error domain name for CloudKit.

6.3.9 CKErrorRetryAfterKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the user dictionary keys for error dictionaries.
Notes: On CKErrorServiceUnavailable or CKErrorRequestRateLimited errors the userInfo dictionary may

contain a number that specifies the period of time in seconds after which the client may retry the request.

6.3.10 CKOwnerDefaultName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The user ID representing the current user. Use this value when creating zone IDs.

6.3.11 CKPartialErrorsByItemIDKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: When a CKErrorPartialFailure happens this key will be set in the error's userInfo dictionary.
Notes: The value of this key will be a dictionary, and the values will be errors for individual items with the keys being the item IDs that failed.

6.3.12 CKRecordChangedErrorAncestorRecordKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the error details keys.
Notes: If the server rejects a record save because it has been modified since the last time it was read, a CKErrorServerRecordChanged error will be returned and it will contain versions of the record in its user-Info dictionary. Apply your custom conflict resolution logic to the server record (CKServerRecordKey) and attempt a save of that record.

6.3.13 CKRecordChangedErrorClientRecordKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the error details keys.
Notes: If the server rejects a record save because it has been modified since the last time it was read, a CKErrorServerRecordChanged error will be returned and it will contain versions of the record in its user-Info dictionary. Apply your custom conflict resolution logic to the server record (CKServerRecordKey) and attempt a save of that record.

6.3.14 CKRecordChangedErrorServerRecordKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the error details keys.
Notes: If the server rejects a record save because it has been modified since the last time it was read, a CKErrorServerRecordChanged error will be returned and it will contain versions of the record in its userInfo dictionary. Apply your custom conflict resolution logic to the server record (CKServerRecordKey) and attempt a save of that record.

6.3.15 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.3.16 Constructor(Container as CKContainerMBS) 219

6.3.16 Constructor(Container as CKContainerMBS)

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Creates a new object for an container reference.
Notes: Allows you to initialize with your subclass to fill events.
See also:

- 6.3.15 Constructor 219

6.3.17 containerWithIdentifier(name as string) as CKContainerMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the container object associated with the specified identifier.
Notes: containerIdentifier: The bundle identifier of the app whose container you want to access. The bundle identifier must be in the app,Äôs com.apple.developer.icloud-container-identifiers entitlement. This parameter must not be nil.

Returns the container object for the designated app,Äôs content, or nil if the container cannot be found.

The specified identifier must correspond to one of the ubiquity containers listed in the iCloud capabilities section of your Xcode project. Including the identifier with your app,Äôs capabilities adds the corresponding entitlements to your app. To access your app,Äôs default container, use the defaultContainer method instead.

6.3.18 defaultContainer as CKContainerMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the default container object for managing the current app,Äôs content.
Notes: Returns the container object associated with the current app,Äôs content. This method always returns a valid container object.

Use this method to retrieve the container for your app,Äôs native content. This container is the one your app typically uses to store its data. If you want the container for a different app, create and initialize that container using the containerWithIdentifier method.

During development, the returned container object points to a development version of your app,Äôs container. When you ship your app, the returned object points to the live production environment.

6.3.19 discoverAllContactUserInfos(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about all discoverable users that are known to the current user.
Notes: Use this method to retrieve information about other users of the app. This method returns information about those users who meet the following criteria:

- There is contact information for the user in the current user,Äôs address book.
- The user has run the app.
- The user has granted the CKApplicationPermissionUserDiscoverability permission to your app for this container.

This method searches for the users asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverAllContactsOperationMBS object and configure the desired priority.

6.3.20 discoverAllIdentities(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches all user records that match an entry in the user,Äôs address book.
Notes: This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverAllContactsOperation object and configure the desired priority.

6.3.21 discoverUserIdentityWithEmailAddress(emailAddress as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the user record ID associated in the user,Äôs contacts with the email address.
Notes: email: The email address used to locate the user.

Use this method to retrieve the ID of a user that is known to the current user. The user you are searching for must meet the following criteria:

- The user must be in the current user,Äôs address book.
- The user must have run the app.
- The user must have granted the userDiscoverability permission for this container.

This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverUserInfosOperationMBS object and configure the desired priority.

6.3.22 discoverUserIdentityWithPhoneNumber(phoneNumber as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the user record ID associated in the user,Äôs contacts with the phone number.
Notes: phoneNumber: The phone number used to locate the user.

Use this method to retrieve the ID of a user that is known to the current user. The user you are searching for must meet the following criteria:

- The user must be in the current user,Äôs address book.
- The user must have run the app.
- The user must have granted the userDiscoverability permission for this container.

This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverUserInfosOperationMBS object and configure the desired priority.

6.3.23 discoverUserIdentityWithUserRecordID(userRecordID as CKRecordIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single user based on the ID of the corresponding user record.
Notes: userRecordID: The ID of the user record.

Use this method to retrieve information about a user for which you already have a user record ID. The user you are searching for must meet the following criteria:

- The user must have run the app.
- The user must have granted the userDiscoverability permission for this container.

This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverUserInfosOperationMBS object and configure the desired priority.

6.3.24 discoverUserInfoWithEmailAddress(emailAddress as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single user based on that user,Äôs email address.
Notes: email: The iCloud email address of the user you want to locate.

Use this method to retrieve the ID of a user that is known to the current user. The user you are searching for must meet the following criteria:

- The user must be in the current user,Äôs address book.
- The user must have run the app.
- The user must have granted the CKApplicationPermissionUserDiscoverability permission for this container.

This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverUserInfosOperationMBS object and configure the desired priority.

6.3.25 discoverUserInfoWithUserRecordID(userRecordID as CKRecordIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single user based on the ID of the corresponding user record.
Notes: userRecordID: The ID of the user record.

Use this method to retrieve information about a user for which you already have a user record ID. The user you are searching for must meet the following criteria:

- The user must have run the app.
- The user must have granted the CKApplicationPermissionUserDiscoverability permission for this container.

This method searches for the user asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKDiscoverUserInfosOperationMBS object and configure the desired priority.

6.3.26 fetchAllLongLivedOperationIDs(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the identifiers of the running or recently completed long-lived operations.
Notes: A long-lived operation is an operation that continues to run after the app exits, described in CKOperationMBS. To get the operation object for an identifier, use the fetchLongLivedOperationWithID method.

6.3.27 fetchLongLivedOperationWithID(operationID as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the running or recently completed long-lived operation specified by the operation identifier.
Notes: operationID: The identifier of the long-lived operation you want to fetch.

A long-lived operation is an operation that continues to run after the app exits, described in CKOperationMBS. To receive the callbacks for a long-lived operations, set its completion block and add it to an operation queue.

6.3.28 fetchShareParticipantWithEmailAddress(emailAddress as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single share participant (a person who accepted a shared record) based on that participant,Äôs email address.
Notes: emailAddress: The iCloud email address of the share participant you want to locate.

This method searches for the share participant asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKFetchShareParticipantsOperationMBS object and configure the desired priority.

6.3.29 fetchShareParticipantWithPhoneNumber(phoneNumber as string, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single share participant (a person who accepted a shared record) based on that participant,Äôs phone number.
Notes: phoneNumber: The phone number used to locate the share participant.

This method searches for the share participant asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKFetchShareParticipantsOperationMBS object and configure the desired priority.

6.3.30 fetchShareParticipantWithUserRecordID(userRecordID as CKRecordIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves information about a single share participant based on the ID of the corresponding user record.
Notes: userRecordID: The ID of the user record.

This method searches for the share participant asynchronously and with a low priority. If you want the task to execute with a higher priority, create a CKFetchShareParticipantsOperationMBS object and configure the desired priority.

6.3.31 fetchUserRecordID(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the user record ID associated with the current user.
Notes: At startup time, fetching the user record ID may take longer while CloudKit makes the initial iCloud account request. After the initial fetch, accessing the user record ID should take less time. If no iCloud account is associated with the device, or if access to the user,Äôs iCloud account is restricted, this method returns an error of type CKErrorNotAuthenticated.

6.3.32 registerCloudKitShare(Share as CKShareMBS, ServiceItems as Variant)

Plugin Version: 21.3, Platform: macOS, Targets: All.

Function: Registers a CloudKit share for the user to modify.
Example:

```
// The following example shows how to create an item provider with an existing share. It then invokes the cloud-sharing service with the provider and presents the share,Äôs configuration to the user.

Public Sub modifyShare(share as CKShareMBS, container as CKContainerMBS)
// Create an item provider and register a share that
// already exists on the server.

Dim itemProvider As New NSSharingServiceItemsMBS
container.registerCloudKitShare(share, itemProvider)

// Create and invoke the cloud-sharing service to
// present the share configuration to the user.
Dim service As NSSharingServiceMBS
service = NSSharingServiceMBS.sharingServiceNamed(NSSharingServiceMBS.NSSharingServiceNameCloud-Sharing)

If service <>Nil Then
If service.canPerformWithItems(itemProvider) then
service.performWithItems(itemProvider)
end if
End If

End Sub
```

Notes: share: The CloudKit share to modify.
self: The CloudKit container that stores the shared records.
ServiceItems: the NSSharingServiceItemsMBS object to add the item provider.

Use this method when the CloudKit share already exists on the server and you want to update it. The behavior of the sharing service depends on the role of the current user. An owner can edit the share,Äôs configuration, which includes managing participants and their permissions. A participant can view the share,Äôs configuration and choose to stop participating.
If you,Äôre unsure which container to use, fetch the share,Äôs metadata using CKFetchShareMetadataOperationMBS. Then initialize an instance of CKContainer using the metadata,Äôs containerIdentifier property.
Use the NSCloudSharingServiceMBS events to respond to any changes the sharing service makes.

To create a new share, use the registerCloudKitShareWithPreparationHandler method instead.

6.3.33 registerCloudKitShareWithPreparationHandler(ServiceItems as Variant, tag as variant = nil)

Plugin Version: 21.3, Platform: macOS, Targets: All.

Function: Registers a handler that prepares a new CloudKit share.
Notes: Calls registerCloudKitShareWithPreparationHandler event to let you fill in code.

ServiceItems: the NSSharingServiceItemsMBS object to add the item provider.
Tag: Variant passed through.

Use this method to share a hierarchy of CloudKit records with other iCloud users. When the service invokes the handler, create an instance of CKShare with a root record. Save the share to the server using CKModifyRecordsOperationMBS. The root record (and its hierarchy) must already exist on the server or be part of the same save operation. After the share saves, call preparationCompletionHandler with the saved share and its container. If the save fails, pass the error to the completion handler instead. Invoking the sharing service with a share you register using this method prompts the user to begin sharing.
Use the NSCloudSharingServiceMBS events to respond to any changes the sharing service makes.
See also:

- 6.3.57 registerCloudKitShareWithPreparationHandler(ServiceItems as Variant, byref share as CKShareMBS, byref container as CKContainerMBS, byref error as NSErrorMBS, tag as variant) 233

6.3.34 requestApplicationPermission(applicationPermission as Integer, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Requests the specified permission from the user to make the user,Äôs identity discoverable.
Notes: applicationPermission: The requested permission. This requested permission applies to the current container only and will not impact permissions granted for other containers. For a list of possible values, see CKApplicationPermissions.

To implement social features in your app, it is possible to correlate a user record with the user,Äôs actual name, but your app must get permission from the user to do so. Making a user record discoverable to the friends (contacts) of that user involves calling the the requestApplicationPermission method and asking for the CKApplicationPermissionUserDiscoverability permission. When you call that method, CloudKit asks the user on your behalf whether the user record should be made discoverable. If the user grants the request, that user,Äôs contacts can discover that user,Äôs true identity when running the app. To discover the contacts of the current user, you use the discoverAllContactUserInfos method or one of several other methods to get the related user information.

The first time you request a permission on any of the user,Äôs devices, the user is prompted to grant or deny the request. Once the user grants or denies a permission, subsequent requests for the same permission (on the same or separate devices), do not prompt the user again.

This method runs asynchronously and delivers the results to the block you provide.

6.3.35 statusForApplicationPermission(applicationPermission as Integer, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Checks the status of the specified permission asynchronously.
Notes: applicationPermission: The permission whose status you want to check. For a list of possible values, see CKApplicationPermissions.

Use this method to determine the extra capabilities granted to your app by the user. If your app has not yet requested a specific permission, calling this method may yield the value CKApplicationPermissionStatusInitialState for the permission. When that value is returned, call the requestApplicationPermission method to request the permission from the user.

6.3.36 Properties

6.3.37 containerIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The string that identifies the app,Äôs container.
Notes: Use the value in this property to distinguish between different container objects in your app. (Read only property)

6.3.38 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.3.39 privateCloudDatabase as CKDatabaseMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database containing the user,Äôs private data.
Notes: The database in this property is available only if the device has an active iCloud account. Access to the database is limited to the user of that iCloud account by default. The current user owns all content in the private database and is allowed to read and write that content. Data in the private database is not visible in the developer portal or to any other users.

Data stored in the private database counts against the storage quota of the current user,Äôs iCloud account.

If there is no active iCloud account on the user,Äôs device, this property still returns a valid database object, but attempts to use that object will return errors. To determine if there is an active iCloud account on the device, use the accountStatus method.
(Read only property)

6.3.40 publicCloudDatabase as CKDatabaseMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database containing the data shared by all users.
Notes: The database in this property is available regardless of whether the user,Äôs device has an active iCloud account. The contents of the public database are readable by all users of the app, and users have write access to the records (and other data objects) they create. Data in the public database is also visible in the developer portal, where you can assign roles to users and restrict access as needed.

Data stored in the public database counts against your app,Äôs iCloud storage quota and not against the quota of any single user.
(Read only property)

6.3.41 sharedCloudDatabase as CKDatabaseMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database containing shared user data.
Notes: The database in this property is available only if the device has an active iCloud account. Permissions on the database are limited to the user based on the permissions of the enclosing CKShareMBS object (representing the record that has been shared). The current user does not own the content in the shared database and is allowed to read and write that content only if that permission has been granted in the CKShareMBS. Data in the shared database is not visible in the developer portal or to any user who has not been granted access.

Data stored in the shared database does not count against the storage quota of the current user,Äôs iCloud account.

If there is no active iCloud account on the user,Äôs device, this property still returns a valid database object, but attempts to use the object will return errors. To determine whether the device has an active iCloud account use the accountStatus method.
(Read only property)

6.3.42 Events

6.3.43 accountStatusCompleted(accountStatus as Integer, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to be called when accountStatus is returned.
Notes: accountStatus: The status of the current user,Äôs iCloud account.
error: An error object, or nil if the status is determined successfully. Use the information in the error object to determine whether the problem has a workaround.

6.3.44 discoverAllContactUserInfosCompleted(userRecordID() as CKDiscoveredUserInfoMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverAllContactUserInfos method.
Notes: Your block must be capable of executing on any thread of the app. This block returns no value and takes the following parameters:

userInfos: An array of CKDiscoveredUserInfoMBS objects. Each object in the array contains information about the user, including the ID of the corresponding user record. If no users are discovered, the provided array is empty.
error: An error object if a problem occurs, or nil if the IDs are retrieved successfully.

6.3.45 discoverAllIdentitiesWithCompleted(userRecordID() as CKUserIdentityMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverAllIdentities method.
Notes: This event returns no value and takes the following parameters:

userIdentities: An array of CKUserIdentityMBS objects that match entries in the user,Äôs contacts.
error: An error object if a problem occurs or nil if the fetch completed successfully.

6.3.46 discoverUserIdentityWithEmailAddressCompleted(emailAddress as String, userInfo as CKUserIdentityMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverUserIdentityWithEmailAddress method.
Notes: This event returns no value and takes the following parameters:

userInfo: The user (CKUserIdentityMBS object) associated with the email address, or nil if an identity could not be located.
error: An error object if a problem occurs or nil if the CKUserIdentityMBS is retrieved successfully.

6.3.47 discoverUserIdentityWithPhoneNumberCompleted(phoneNumber as String,userInfo as CKUserIdentityMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverUserIdentityWithPhoneNumber method.
Notes: This event returns no value and takes the following parameters:

userInfo: The user (CKUserIdentityMBS object) associated with the phone number, or nil if an identity could not be located.
error: An error object if a problem occurs, or nil if the CKUserIdentityMBS is retrieved successfully.

6.3.48 discoverUserIdentityWithUserRecordIDCompleted(userRecordID as CKRecordIDMBS, userInfo as CKUserIdentityMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverUserIdentityWithUserRecordID.
Notes: This event returns no value and takes the following parameters:

userInfo
An object containing information about the user, or nil if the user is not found.

error
An error object if a problem occurs, or nil if the ID is retrieved successfully.

6.3.49 discoverUserInfoWithEmailAddressCompleted(emailAddress as String, userInfo as CKDiscoveredUserInfoMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for discoverUserInfoWithEmailAddress method.
Notes: This event returns no value and takes the following parameters:

userInfo: An object containing information about the user or nil if the user is not found.
error: An error object if a problem occurs or nil if the ID is retrieved successfully.

6.3.50 discoverUserInfoWithUserRecordIDCompleted(userRecordID as CKRecordIDMBS, userInfo as CKDiscoveredUserInfoMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the results for discoverUserInfoWithUserRecordID method.
Notes: This event returns no value and takes the following parameters:

userInfo: An object containing information about the user or nil if the user is not found.
error: An error object if a problem occurs or nil if the ID is retrieved successfully.

6.3.51 fetchAllLongLivedOperationIDsCompleted(outstandingOperationIDs() as String, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called for fetchAllLongLivedOperationIDs method.
Notes: outstandingOperationsByIDs: An array containing the identifiers for all the active long-lived opera-

tions. If a long-lived operation is canceled or completed, it is no longer an active operation, and its identifier will not be included in this array. An operation is complete if the app successfully receives the completion callback.
error: An error object, or nil if the fetch is successful.

6.3.52 fetchLongLivedOperationWithIDCompleted(operationID as String, outstandingOperation as CKOperationMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called when fetchLongLivedOperationWithID finished.
Notes: outstandingOperation: The proxy object for the corresponding long-lived operation. If a long-lived operation was canceled or completed, this is nil.
error: An error object, or nil if the fetch is successful.

6.3.53 fetchShareParticipantWithEmailAddressCompleted(emailAddress as String, shareParticipant as CKShareParticipantMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchShareParticipantWithEmailAddress method.
Notes: This event returns no value and takes the following parameters:

shareParticipant: An object containing information about the share participant, or nil if the participant is not found.
error: An error object if a problem occurs, or nil if the participant is retrieved successfully.

6.3.54 fetchShareParticipantWithPhoneNumberCompleted(phoneNumber as String,shareParticipant as CKShareParticipantMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchShareParticipantWithPhoneNumber method.
Notes: shareParticipant: An object containing information about the share participant, or nil if the participant is not found.
error: An error object if a problem occurs, or nil if the participant is retrieved successfully.

6.3.55 fetchShareParticipantWithUserRecordIDCompleted(userRecordID as CKRecordIDMBS, shareParticipant as CKShareParticipantMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called for fetchShareParticipantWithUserRecordID.
Notes: This event returns no value and takes the following parameters:

shareParticipant: An object containing information about the share participant, or nil if the participant is not found.
error: An error object if a problem occurs, or nil if the participant is retrieved successfully.

6.3.56 fetchUserRecordIDCompleted(userRecordID as CKRecordIDMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchUserRecordID method.
Notes: recordID: The ID of the user record for the current user, or nil if the current device is not configured with an iCloud account.
error: An error object if a problem occurs, or nil if the record ID is retrieved successfully.

6.3.57 registerCloudKitShareWithPreparationHandler(ServiceItems as Variant, byref share as CKShareMBS, byref container as CKContainerMBS, byref error as NSErrorMBS, tag as variant)

Plugin Version: 21.3, Platform: macOS, Targets: .

Function: The event called by registerCloudKitShareWithPreparationHandler method.
Notes: ServiceItems: the NSSharingServiceItemsMBS object to add the item provider.
Tag: Variant passed through.
share: The share to use.
container: The container to use.
error: The error you encountered.
See also:

- 6.3.33 registerCloudKitShareWithPreparationHandler(ServiceItems as Variant, tag as variant = nil) 226

6.3.58 requestApplicationPermissionCompleted(applicationPermissionStatus as Integer, accountStatus as Integer, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for requestApplicationPermission method.

6.3.59 statusForApplicationPermissionCompleted(applicationPermissionStatus as Integer, accountStatus as Integer, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for statusForApplicationPermission method.

6.3.60 Constants

Constants

Constant	Value	Description
CKApplicationPermissionStatusCouldNotComplete	1	One of the constants indicating whether the app has been permission. An error occurred during the getting or setting of the app p the provided NSErrorMBS object for more information.
CKApplicationPermissionStatusDenied	2	One of the constants indicating whether the app has been permission. The user denied access to the permission.
CKApplicationPermissionStatusGranted	3	One of the constants indicating whether the app has been permission. The user granted access to the permission.
CKApplicationPermissionStatusInitialState	0	One of the constants indicating whether the app has been permission. The user has not yet decided whether to grant this permi questApplicationPermission method to ask the user to gran
CKApplicationPermissionUserDiscoverability	1	The current user is discoverable (through the user,Äôs emai users of the app.

Status

Constant	Value	Description
CKAccountStatusAvailable	1	The user,Äôs iCloud account is available and may be used by this app.
CKAccountStatusCouldNotDetermine	0	An error occurred during an attempt to retrieve the account status. Co the provided NSError object for more information.
CKAccountStatusNoAccount	3	The user,Äôs iCloud account is not available because no account inform has been provided for this device.
CKAccountStatusRestricted	2	The user,Äôs iCloud account is not available. Access was denied due to Pa Controls or Mobile Device Management restrictions.

Errors

Constant	Value	Description
ErrorAlreadyShared	30	A record/share cannot be saved, doing so would cause a hierarchy of rec to exist in multiple shares.
ErrorAssetFileModified	17	Asset file content was modified while being saved.
ErrorAssetFileNotFound	16	Asset file was not found.
ErrorBadContainer	5	Un-provisioned or unauthorized container. Try provisioning the containe fore retrying the operation.
ErrorBadDatabase	24	Operation could not be completed on the given database. Likely cause attempting to modify zones in the public database.
ErrorBatchRequestFailed	22	One of the items in this batch operation failed in a zone with atomic upd so the entire batch was rejected.
ErrorChangeTokenExpired	21	The previousServerChangeToken value is too old and the client must re- from scratch.
ErrorConstraintViolation	19	The server rejected the request because there was a conflict with a unique
ErrorIncompatibleVersion	18	App version is less than the minimum allowed version.
ErrorInternalError	1	CloudKit.framework encountered an error. This is a non-recoverable erro
ErrorInvalidArguments	12	Bad client request (bad record graph, malformed predicate)
ErrorLimitExceeded	27	The request to the server was too large. Retry this request as a smaller b
ErrorManagedAccountRestricted	32	Request was rejected due to a managed account restriction.
ErrorMissingEntitlement	8	Missing entitlement
ErrorNetworkFailure	4	Network error (available but CFNetwork gave us an error)
ErrorNetworkUnavailable	3	Network not available
ErrorNotAuthenticated	9	Not authenticated (writing without being logged in, no user record)
ErrorOperationCancelled	20	A CKOperation was explicitly cancelled.
ErrorPartialFailure	2	Some items failed, but the operation succeeded overall. Check CKPartia rorsByItemIDKey in the userInfo dictionary for more details.
ErrorParticipantMayNeedVerification	33	Share Metadata cannot be determined, because the user is not a memb the share. There are invited participants on the share with email addr or phone numbers not associated with any iCloud account. The user ma able to join the share if they can associate one of those email address phone numbers with their iCloud account via the system Share Accept Call UIApplication's openURL on this share URL to have the user attem verify their information.
ErrorPermissionFailure	10	Access failure (save, fetch, or shareAccept)
ErrorQuotaExceeded	25	Saving a record would exceed quota
ErrorReferenceViolation	31	The target of a record's parent or share reference was not found.
ErrorRequestRateLimited	7	Client is being rate limited
ErrorResultsTruncated	13	Results Truncated. Deprecated and will not be returned.
ErrorServerRecordChanged	14	The record was rejected because the version on the server was different.
ErrorServerRejectedRequest	15	The server rejected this request. This is a non-recoverable error.
ErrorServiceUnavailable	6	Service unavailable
ErrorTooManyParticipants	29	A share cannot be saved because there are too many participants attache the share.
ErrorUnknownItem	11	Record does not exist
ErrorUserDeletedZone	28	The user deleted this zone through the settings UI. Your client should e remove its local data or prompt the user before attempting to re-upload data to this zone.
ErrorZoneBusy	23	The server is too busy to handle this zone operation. Try the operation a in a few seconds.
ErrorZoneNotFound	26	The specified zone does not exist on the server.

6.4 class CKDatabaseMBS

6.4.1 class CKDatabaseMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A conduit for accessing and for performing operations on the public and private data of an app container.
Notes: An app container has a public database whose data is accessible to all users and a private database whose data is accessible only to the current user. A database object takes requests for data and applies them to the appropriate part of the container.

You do not create database objects yourself, nor should you subclass CKDatabaseMBS. Your app,Äôs CKContainerMBS objects provide the CKDatabaseMBS objects you use to access the associated data. Use database objects as-is to perform operations on data.

The public database is always available, regardless of whether the device has an an active iCloud account. When no iCloud account is available, your app may fetch records and perform queries on the public database, but it may not save changes. (Saving records to the public database requires an active iCloud account to identify the owner of those records.) Access to the private database always requires an active iCloud account on the device.

see
https://developer.apple.com/reference/cloudkit/ckdatabase
Blog Entries

- MBS Xojo Plugins, version 21.3pr3
- MBS Xojo Plugins, version 21.2pr3

6.4.2 Methods

6.4.3 addOperation(operation as CKDatabaseOperationMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Executes the specified operation asynchronously against the current database.
Notes: operation: The operation object to execute. You must configure the operation object with any dependencies and completion handlers before calling this method. If this parameter is nil, the method does nothing.

Do not change the properties of the operation object after calling this method. Prior to executing the operation, this method sets the operation object,Äôs database property to the current database, replacing any previously assigned database.

This method executes the operation object with the priority you assigned to the object through its queuePriority property.

6.4.4 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.4.5 Constructor(Container as CKContainerMBS, databaseScope as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Retrieves the database with the appropriate scope.
Notes: databaseScope: The scope of the database you want returned.

Returns an initialized CKDatabase object with the appropriate scope.
This convenience method returns the database associated with the specified container that is of the scope requested.
See also:

- 6.4.6 Constructor(Database as CKDatabaseMBS) 238

6.4.6 Constructor(Database as CKDatabaseMBS)

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Creates a new object for an database reference.
Notes: Allows you to initialize with your subclass to fill events.
See also:

- 6.4.5 Constructor(Container as CKContainerMBS, databaseScope as Integer) 238

6.4.7 deleteRecordWithID(recordID as CKRecordIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Deletes the specified record asynchronously, with a low priority, from the current database.
Notes: recordID: The ID of the record you want to delete. This method throws an exception if this parameter is nil.

Deleting a record may trigger additional deletions if the record was referenced by other records. This method reports only the ID of the record you asked to delete. CloudKit does not report deletions triggered by owning relationships between records.

This method deletes the record with a low priority, which may cause the task to execute after higher-priority tasks. To delete records more urgently, create a CKModifyRecordsOperationMBS object with the desired priority. You can also use that operation object to delete multiple records simultaneously.

6.4.8 deleteRecordWithIDSync(recordID as CKRecordIDMBS, byref error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Deletes the specified record synchronously, with a low priority, from the current database.
Notes: recordID: The ID of the record you want to delete. This method throws an exception if this parameter is nil.

Deleting a record may trigger additional deletions if the record was referenced by other records. This method reports only the ID of the record you asked to delete. CloudKit does not report deletions triggered by owning relationships between records.

This method deletes the record with a low priority, which may cause the task to execute after higher-priority tasks. To delete records more urgently, create a CKModifyRecordsOperationMBS object with the desired priority. You can also use that operation object to delete multiple records simultaneously.

6.4.9 deleteRecordZone(zoneID as CKRecordZoneIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Deletes one record zone (and its contents) asynchronously, with a low priority, from the current database.
Notes: zoneID: The ID of the zone you want to delete. This method throws an exception if this parameter is nil.

Deleting a zone permanently deletes the zone and all records in that zone. After deleting the zone, you can use the same ID to create a new empty zone.

This method deletes the record zone with a low priority, which may cause the task to execute after higher-priority tasks. To delete the record zone more urgently, create a CKModifyRecordZonesOperationMBS object with the desired priority. You can also use that operation object to delete multiple record zones

simultaneously.

6.4.10 deleteSubscriptionWithID(subscriptionID as String, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Deletes one subscription object asynchronously, with a low priority, from the current database.
Notes: subscriptionID: The ID of the subscription object to delete. This method throws an exception if this parameter is nil.

Deleting a subscription stops the subscription from watching its changed records and sending alerts.

This method deletes the subscription object with a low priority, which may cause the task to execute after higher-priority tasks. To delete the subscription more urgently, create a CKModifySubscriptionsOperationMBS object with the desired priority. You can also use that operation object to delete multiple subscription objects simultaneously.

6.4.11 fetchAllRecordZones(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches all record zones asynchronously, with a low priority, from the current database.
Notes: Use this method to locate the record zones in this database. Record zones represent groups of records with a common purpose.

This method fetches record zones with a low priority, which may cause the task to execute after higher-priority tasks. To fetch record zones more urgently, create a CKFetchRecordZonesOperationMBS object with the desired priority.

6.4.12 fetchAllSubscriptions(tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches all subscription objects asynchronously, with a low priority, from the current database.
Notes: Use this method to locate the subscriptions in the current database. Subscriptions represent persistent queries on the server. A subscription can be used to alert the app when records change.

This method fetches the subscription objects with a low priority, which may cause the task to execute after higher-priority tasks. To fetch subscriptions more urgently, create a CKFetchSubscriptionsOperationMBS

object with the desired priority.

6.4.13 fetchRecordWithID(recordID as CKRecordIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches one record asynchronously, with a low priority, from the current database.
Notes: recordID: The ID of the record you want to fetch. This method throws an exception if this parameter is nil.

Use this method to fetch records that are not urgent to your app,Äôs execution. This method fetches the record with a low priority, which may cause the fetch to execute after higher-priority tasks. To fetch records more urgently, create a CKFetchRecordsOperation object with the desired priority. You can also use that operation object to fetch multiple records simultaneously.

6.4.14 fetchRecordWithIDSync(recordID as CKRecordIDMBS, byref record as CKRecordMBS, byref error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches one record synchronously, with a low priority, from the current database.
Notes: recordID: The ID of the record you want to fetch. This method throws an exception if this parameter is nil.
record: The requested record object. If no such record is found, this parameter is nil.
error: An error object, or nil if the record was fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

Use this method to fetch records that are not urgent to your app,Äôs execution. This method fetches the record with a low priority, which may cause the fetch to execute after higher-priority tasks. To fetch records more urgently, create a CKFetchRecordsOperation object with the desired priority. You can also use that operation object to fetch multiple records simultaneously.

6.4.15 fetchRecordZoneWithID(zoneID as CKRecordZoneIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches one record zone asynchronously, with a low priority, from the current database.
Notes: zoneID: The ID of the record zone. This method throws an exception if this parameter is nil.

Use this method to retrieve a record zone whose ID you already know. You might retrieve a record zone object so that you can assess its capabilities.

This method fetches the record zone with a low priority, which may cause the task to execute after higher-priority tasks. To fetch the record zone more urgently, create a CKFetchRecordZonesOperationMBS object with the desired priority. You can also use that operation object to fetch multiple record zones simultaneously.

6.4.16 fetchSubscriptionWithID(subscriptionID as String, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Fetches one subscription object asynchronously, wiht a low priority, from the current database.
Notes: subscriptionID: The ID of the subscription object. This method throws an exception if this parameter is nil.

Use this method to retrieve a subscription object whose ID you already know. You might retrieve a subscription object so that you can assess its attributes or update the notification information used to generate alerts.

This method fetches the subscription object with a low priority, which may cause the task to execute after higher-priority tasks. To fetch the subscription more urgently, create a CKFetchSubscriptionsOperationMBS object with the desired priority. You can also use that operation object to fetch multiple subscription objects simultaneously.

6.4.17 performQuery(query as CKQueryMBS, zoneID as CKRecordZoneIDMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Searches the specified zone asynchronously for records that match the query parameters.
Notes: query: The query object containing the parameters for the search. This method throws an exception if this parameter is nil. For information about how to construct queries, see CKQueryMBS.
zoneID: The ID of the zone to search. Search results are limited to records in the specified zone. Specify nil to search the default zone of the database.

Use this method to execute searches against the current database. Do not use this method when the number of returned records is potentially more than a few hundred records; when more records are needed, create an execute a CKQueryOperationMBS instead of calling performQuery:inZoneWithID on the CKDatabaseMBS. For efficiency, all queries automatically limit the number of returned records based on current conditions. If your query hits the maximum value, this method returns only the first portion of the overall results. The number of returned records should be sufficient in most cases, but to get the entire set of records you must create and execute a CKQueryOperationMBS object instead. Query operations also return a maximum number of results, but when they do, they provide a cursor object that you can use to fetch the next batch

of results.

You can search any content that is represented by a CKRecord object, including user records. You cannot use this method to search for CKSubscriptionMBS or CKRecordZoneMBS objects.

6.4.18 saveRecord(record as CKRecordMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Saves one record asynchronously, with a low priority, to the current database, if the record has never been saved or if it is newer than the version on the server.
Notes: record: The record to save. This method throws an exception if this parameter is nil.

This method saves the record only if it has never been saved before or if it is newer than the version on the server. You cannot use this method to overwrite newer versions of a record on the server.

This method saves the record with a low priority, which may cause the task to execute after higher-priority tasks. To save records more urgently, create a CKModifyRecordsOperationMBS object with the desired priority. You can also use that operation object to save multiple records simultaneously.

6.4.19 saveRecordSync(record as CKRecordMBS, byref error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Saves one record synchronously, with a low priority, to the current database, if the record has never been saved or if it is newer than the version on the server.
Notes: record: The record to save. This method throws an exception if this parameter is nil.

This method saves the record only if it has never been saved before or if it is newer than the version on the server. You cannot use this method to overwrite newer versions of a record on the server.

This method saves the record with a low priority, which may cause the task to execute after higher-priority tasks. To save records more urgently, create a CKModifyRecordsOperationMBS object with the desired priority. You can also use that operation object to save multiple records simultaneously.

6.4.20 saveRecordZone(zone as CKRecordZoneMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Saves one record zone asynchronously, with a low priority, to the current database.

Notes: zone: The zone you want to save to the database. This method throws an exception if this parameter is nil.

Use this method to save a record zone to the database so that you can subsequently store records in it. Record zones must be saved before you attempt to save any records that reside in that zone. Because this method executes asynchronously, use the completion handler to verify that the zone was saved successfully before attempting to save any records.

This method saves the record zone with a low priority, which may cause the task to execute after higher-priority tasks. To save the record zone more urgently, create a CKModifyRecordZonesOperationMBS object with the priority you want. You can also use that operation object to save multiple record zones all at once.

6.4.21 saveSubscription(subscription as CKSubscriptionMBS, tag as Variant = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Saves one subscription object asynchronously, with a low priority, to the current database.
Notes: subscription: The subscription object you want to save to the database. This method throws an exception if this parameter is nil.

Use this method to save a subscription to the database so that the subsription can begin watching for changes.

This method saves the subscription object with a low priority, which may cause the task to execute after higher-priority tasks. To save the subscription more urgently, create a CKModifySubscriptionsOperation-MBS object with the desired priority. You can also use that operation object to save multiple subscription objects simultaneously.

6.4.22 Properties

6.4.23 databaseScope as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of database (public, private, or shared).
Notes: (Read only property)

6.4.24 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.4.25 Events

6.4.26 deleteRecordWithIDCompleted(recordID as CKRecordIDMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the results for deleteRecordWithID method.
Notes: recordID: The ID of the record you attempted to delete.
error: An error object, or nil if the record zone was deleted successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.27 deleteRecordZoneWithIDCompleted(zoneID as CKRecordZoneIDMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the results for deleteRecordZone method.
Notes: zoneID: The ID of the zone that you tried to delete.
error: An error object, or nil if the record zone was deleted successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.28 deleteSubscriptionWithIDCompleted(subscriptionID as String, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for deleteSubscriptionWithID method.
Notes: subscriptionID: The ID of the subscription object you attempted to delete.
error: An error object, or nil if the subscription was deleted successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.29 fetchAllRecordZonesCompleted(zones() as CKRecordZoneMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchAllRecordZones method.
Notes: zones: An array of CKRecordZoneMBS objects. The returned array always contains at least one record zone corresponding to the default zone.
error: An error object, or nil if the record zones were fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.30 fetchAllSubscriptionsCompleted(subscriptions() as CKSubscriptionMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchAllSubscriptions.
Notes: subscriptions: An array of CKSubscription objects. If the subscription objects cannot be retrieved, the array is empty.
error: An error object, or nil if the subscriptions were fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.31 fetchRecordWithIDCompleted(recordID as CKRecordIDMBS, record as CKRecordMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results.
Notes: record: The requested record object. If no such record is found, this parameter is nil.
error: An error object, or nil if the record was fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.32 fetchRecordZoneWithIDCompleted(zoneID as CKRecordZoneIDMBS, zone as CKRecordZoneMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchRecordZoneWithID method.
Notes: zone: The requested CKRecordZone object or nil if the object is not found or cannot be retrieved.
error: An error object, or nil if the record zone was fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.33 fetchSubscriptionWithIDCompleted(subscriptionID as String, subscription as CKSubscriptionMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for fetchSubscriptionWithID.
Notes: subscription: The requested subscription object or nil if the object is not found or cannot be retrieved.
error: An error object, or nil if the subscription was fetched successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.34 performQueryCompleted(query as CKQueryMBS, zoneID as CKRecordZoneIDMBS, results() as CKRecordMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the search results.
Notes: The event to execute with the search results for performQuery method.

results: An array containing zero or more CKRecord objects. The returned records correspond to the records in the specified zone that match the parameters of the query.

error: An error object, or nil if the query was completed successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.35 saveRecordCompleted(record as CKRecordMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for saveRecord.
Notes: record: The record object you attempted to save.
error: An error object, or nil if the record was saved successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.36 saveRecordZoneCompleted(zone as CKRecordZoneMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results for saveRecordZone method.
Notes: zone: The CKRecordZone object you attempted to save.
error: An error object, or nil if the record zone was saved successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.37 saveSubscriptionCompleted(subscription as CKSubscriptionMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the results for saveSubscription method.
Notes: subscription: The subscription object you attempted to save.
error: An error object, or nil if the subscription was saved successfully. Use the information in the error object to determine whether a problem has a workaround.

6.4.38 Constants

Database Scopes

Constant	Value	Description
ScopePrivate	2	The private database.
ScopePublic	1	The public database.
ScopeShared	3	The shared database.

6.5 class CKDatabaseNotificationMBS

6.5.1 class CKDatabaseNotificationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A notification object about a database.
Notes: Subclass of the CKNotificationMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.5.2 Methods

6.5.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.5.4 Properties

6.5.5 databaseScope as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database scope.
Notes: (Read only property)

6.6 class CKDatabaseOperationMBS

6.6.1 class CKDatabaseOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The CKDatabaseOperation class is an abstract parent class for operations that act on the public or private databases in a container.
Notes: Database operations typically involve fetching and saving records and other database objects, as well as queries on the contents of the database. You use the property of this class to tell the operation object which database to use when executing its task. Do not subclass this class or create instances of it. Instead, create instances of one of its concrete subclasses.

Available on macOS 10.12 or newer.
Subclass of the CKOperationMBS class.

6.6.2 Methods

6.6.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.6.4 Properties

6.6.5 database as CKDatabaseMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database that is the target of the operation.
Notes: For operations you plan to execute from your own custom queue, use this property to specify the target database. Setting the database also sets the corresponding container, inherited from the CKOperationMBS parent class. If the value of this property is nil, the operation targets the private database of the app,Äôs default container.

The default value of this property is nil.
(Read and Write property)

6.7 class CKDatabaseSubscriptionMBS

6.7.1 class CKDatabaseSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A subscription for database changes.
Notes: Subclass of the CKSubscriptionMBS class.

6.7.2 Methods

6.7.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.7.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.7.5 Constructor(subscriptionID as string) 251

6.7.5 Constructor(subscriptionID as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.7.4 Constructor 251

6.7.6 copy as CKDatabaseSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.7.7 Properties

6.7.8 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type.
Notes: (Read and Write property)

6.8 class CKDiscoverAllContactsOperationMBS

6.8.1 class CKDiscoverAllContactsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKDiscoverAllUserIdentitiesOperationMBS instead. **Function:** A CKDiscoverAllContactsOperation object retrieves the IDs of all discoverable users that are also in the user,Äôs address book.
Notes: The search of the user,Äôs address book does not return any personal data about the user,Äôs contacts. The search returns only the IDs of the corresponding user records, which contain only data that your app puts there. CloudKit uses the address book information to identify users of the app that the current user knows. Because the system accesses the address book instead of your app, the system does not display a prompt to the user when that access occurs.

Users of an app must opt in to discoverability before their user records can be accessed. To opt in for a user, your app must call the requestApplicationPermission:completion: method of your container object and request the userDiscoverability permission. Calling that method prompts the user to grant or deny the permission.

If you assign a completion block to the completionBlock property of the operation object, the completion block is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.

No Address Book access dialog will be displayed.
Available in OS X 10.10 and newer. Deprecated in 10.12 in favor of CKDiscoverAllUserIdentitiesOperationMBS classes.
Subclass of the CKOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.3pr7

6.8.2 Methods

6.8.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.8.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.8.5 Events

6.8.6 discoverAllContactsCompleted(userInfos() as CKDiscoveredUserInfoMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to use to process the search results.
Notes: The event has no return value and takes the following parameters:

userRecordIDs: An array of CKDiscoveredUserInfoMBS objects. Each ID corresponds to a user of the app who opts in to discovery by other known users. Use this ID to retrieve the corresponding user record.
operationError: An error object containing information about a problem, or nil if the user record IDs are retrieved successfully.

This event is executed only once, at the conclusion of the operation. If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

6.9 class CKDiscoverAllUserIdentitiesOperationMBS

6.9.1 class CKDiscoverAllUserIdentitiesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that finds all discoverable users in the device,Äôs contacts.
Notes: The operation will not generate a dialog indicating that the contacts are being accessed.

Available in OS X 10.12 and newer.
Subclass of the CKOperationMBS class.

6.9.2 Methods

6.9.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to search the device,Äôs contacts.
Notes: You can use the returned CKDiscoverAllUserIdentitiesOperationMBS only once. When executed, this query object performs a new search and returns a batch of results.

6.9.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.9.5 Events

6.9.6 discoverAllUserIdentitiesCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results.
Notes: operationError: An error object that contains the information about a problem or nil if the results are retrieved successfully.

This event is executed only once and represents your last chance to process the operation results. It is executed after all of the individual progress blocks but before the operation,Äôs completed event. The block is executed serially with respect to the other progress blocks of the operation. If you intend to use this

block to process results, update the value of this property before executing the operation or submitting the operation object to a queue.

6.9.7 userIdentityDiscovered(identity as CKUserIdentityMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each user identity returned.
Notes: identity: A CKUserIdentityMBS object that was discovered in the device,Äôs contacts.

This event is executed once for each identity that is discovered. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

6.10 class CKDiscoveredUserInfoMBS

6.10.1 class CKDiscoveredUserInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKUserIdentityMBS instead. **Function:** A CKDiscoveredUserInfo object contains information about a discoverable user in a database.
Notes: You do not create instances of this class yourself. Instead, use a CKDiscoverAllContactsOperation object or other means to retrieve these objects.

Available in OS X 10.10 and newer.
Deprecated in OS X 10.12 in favor of CKUserIdentityMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 20.3pr7

6.10.2 Methods

6.10.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.10.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.10.5 Properties

6.10.6 displayContact as Variant

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Contains the given and family name of the user who is not associated with the local Address Book.

Notes: DisplayContact is not associated with the local Address Book. It is a wrapper around information known to the CloudKit server, including first and last names.
Value is a CNContactMBS object.
(Read only property)

6.10.7 firstName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The first name of the user.
Notes: (Read only property)

6.10.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.10.9 lastName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The last name of the user.
Notes: (Read only property)

6.10.10 userRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the user record.
Notes: Use this value to retrieve the user record associated with the specified user. The user record does not contain any personal information about the user by default. Your app can add data to the user record but should not add any sensitive user data to it.
(Read only property)

6.11 class CKDiscoverUserIdentitiesOperationMBS

6.11.1 class CKDiscoverUserIdentitiesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that fetches user identities.
Notes: Subclass of the CKOperationMBS class.

6.11.2 Methods

6.11.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to discover user identities.
Notes: You can use the returned CKDiscoverUserIdentitiesOperation only once. When executed, this query object processes the share metadata.
See also:

- 6.11.4 Constructor(userIdentityLookupInfos() as CKUserIdentityLookupInfoMBS) 259

6.11.4 Constructor(userIdentityLookupInfos() as CKUserIdentityLookupInfoMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to discover all user identities associated with the specified lookup info.
Notes: userIdentityLookupInfos: An array of CKUserIdentityLookupInfoMBS objects. This parameter is used to initialize the value in the userIdentityLookupInfos property. If you specify nil, you must assign an appropriate value to the userIdentityLookupInfos property before executing the operation.

Returns an initialized operation object.
See also:

- 6.11.3 Constructor 259

6.11.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.11.6 setUserIdentityLookupInfos(IDs() as CKUserIdentityLookupInfoMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the lookup info (email address, phone number, or record ID) used to discover user identities.
Notes: Use this property to view or change the lookup info used to discover user identities. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.11.7 userIdentityLookupInfos as CKUserIdentityLookupInfoMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The lookup info (email address, phone number, or record ID) used to discover user identities.
Notes: Use this property to view or change the lookup info used to discover user identities. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.11.8 Events

6.11.9 discoverUserIdentitiesCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute when the operation has completed.
Notes: This block is called when the operation completes. The completionBlock of the underlying NSOperationMBS is also called if both are set. This block is executed serially with respect to the other progress blocks of the operation. If you intend to use this block to process results, update the value of this property before executing the operation or submitting the operation object to a queue.

6.11.10 userIdentityDiscovered(identity as CKUserIdentityMBS, lookupInfo as CKUserIdentityLookupInfoMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each user identity that has been discovered.
Notes: Each time the event is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.12 class CKDiscoverUserInfosOperationMBS

6.12.1 class CKDiscoverUserInfosOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKDiscoverUserIdentitiesOperationMBS instead. **Function:** A CKDiscoverUserInfosOperation object retrieves the IDs for discoverable users whose email addresses or user record IDs you already know.
Notes: This operation returns information about the corresponding users.

Users of an app must opt in to discoverability before their user records can be accessed. To opt in for a user, your app must call the requestApplicationPermission:completion: method of your container object and request the userDiscoverability permission. Calling that method prompts the user to grant or deny the permission.

If you assign a completed event of the operation object, the completed event is called after the operation executes and returns its results to you. You can use a completed event to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completed event you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.3pr7

6.12.2 Methods

6.12.3 Constructor(emailAddresses() as String, userRecordIDs() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes the operation object with the specified email addresses.
Notes: emailAddresses: An array of strings, each of which contains a single email address to look for. The value you specify is used to initialize the value of the emailAddresses property. If you specify nil for this parameter, assign at least one email address to the emailAddresses property before executing the operation object.
userRecordIDs: An array of CKRecordIDMBS objects, each of which contains the ID of a user record to look for. The value you specify is used to initialize the value of the userRecordIDs property. If you specify nil for this parameter, assign at least one email address to the userRecordIDs property before executing the operation object.

The email addresses you specify need not belong to users in the current user,Äôs address book. The users associated with those email addresses must be discoverable for their user record ID to be returned.

6.12.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.12.5 emailAddresses as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The email addresses of the users whose information you want to retrieve.
Notes: This property contains an array of strings, each of which corresponds to a single email address. Use this property to add or remove email addresses to the operation. If you intend to change the property value, do so before executing the operation or submitting the operation object to a queue.

The default value of this property is set to the email addresses you specify in the Constructor.

6.12.6 setEmailAddresses(emails() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the email addresses of the users whose information you want to retrieve.
Notes: This property contains an array of strings, each of which corresponds to a single email address. Use this property to add or remove email addresses to the operation. If you intend to change the property value, do so before executing the operation or submitting the operation object to a queue.

The default value of this property is set to the email addresses you specify in the Constructor.

6.12.7 setUserRecordIDs(IDs() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the users whose information you want to retrieve.
Notes: This property contains an array of CKRecordIDMBS objects, each of which corresponds to the ID of a user record. Use this property to add or remove record IDs to the operation. If you intend to change the value of this property, you must do so before executing the operation or submitting the operation object to a queue.

The default value of this property is set to the record IDs you specify in the Constructor.

6.12.8 userRecordIDs as CKRecordIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the users whose information you want to retrieve.
Notes: This property contains an array of CKRecordIDMBS objects, each of which corresponds to the ID of a user record. Use this property to add or remove record IDs to the operation. If you intend to change the value of this property, you must do so before executing the operation or submitting the operation object to a queue.

The default value of this property is set to the record IDs you specify in the Constructor.

6.12.9 Events

6.12.10 discoverUserInfosCompleted(emailsToUserInfos as Dictionary, userRecordIDsToUserInfos as Dictionary, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to use to process the search results.
Notes: The event has no return value and takes the following parameters:

emailsToUserInfos: A dictionary whose keys are events with the email addresses you provided in the emailAddresses property. The value of each key is a CKDiscoveredUserInfo object that identifies the user associated with the email address. Use that object to get the user,Äôs name and fetch the corresponding user record.
userRecordIDsToUserInfos: A dictionary whose keys are the CKRecordIDMBS objects you provided. The value of each key is a CKDiscoveredUserInfo object that identifies the user associated with the user record. You can use that object to get the user,Äôs name.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

This event is executed only once, at the conclusion of the operation. If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

6.13 class CKFetchDatabaseChangesOperationMBS

6.13.1 class CKFetchDatabaseChangesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that fetches changes to the database.
Notes: Subclass of the CKDatabaseOperationMBS class.

6.13.2 Methods

6.13.3 Constructor(previousServerChangeToken as CKServerChangeTokenMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured with the previous change token.
Notes: previousServerChangeToken: The previous change token.

Returns an initialized operation object.

After initializing the operation, assign a block to the fetchDatabaseChangesCompletionBlock property to process the results.

If a change anchor from a previous CKFetchDatabaseChangesOperation is passed in, only the zones that have changed since that anchor will be returned.

This per-database CKServerChangeToken is not to be confused with the per-recordZone CKServerChange-Token from CKFetchRecordZoneChangesOperationMBS.

If this is your first fetch or if you wish to re-fetch all zones, pass nil for the change token. Change tokens are opaque and clients should not infer any behavior based on their content.

6.13.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.13.5 Properties

6.13.6 fetchAllChanges as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Configuration setting to control whether this operation makes repeated calls to the server.
Notes: When this property is set to true, the operation sends repeated requests to the server until all record zone changes have been fetched. When this property is set to false, it is the responsibility of the caller to issue subsequent fetch-change operations when moreComing is true in a fetchDatabaseChangesCompleted invocation.

fetchAllChanges is true by default.
(Read and Write property)

6.13.7 previousServerChangeToken as CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The change token from a previous fetch operation.
Notes: Assign the token you received previously from the fetchDatabaseChangesCompleted event to this property. Doing so yields only the changes that have occurred since your last fetch operation. If you specify nil for this parameter, the operation starts fetching changes at the beginning.
(Read and Write property)

6.13.8 resultsLimit as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The maximum number of results to fetch with this operation.
Notes: Use this property to limit the number of changes processed in one operation. When the results limit is reached, the operation object updates the original token you provided and returns it to you to let you know that more results are available.
(Read and Write property)

6.13.9 Events

6.13.10 changeTokenUpdated(serverChangeToken as CKServerChangeTokenMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the change token has changed.

Notes: This event will be executed periodically, giving you an updated change token so that already-fetched record zone changes do not need to be re-fetched on a subsequent operation.

6.13.11 fetchDatabaseChangesCompleted(serverChangeToken as CKServerChangeTokenMBS, moreComing as Boolean, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the operation completes.
Notes: serverChangeToken: The current server change token to be stored and used in subsequent CKFetchDatabaseChangesOperation instances.
moreComing: Indicates if this is the last record zone change. If fetchAllChanges is false, it is the responsibility of the client to create additional CKFetchDatabaseChangesOperation instances for the additional changes.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The client is responsible for saving the change token at the end of the operation and passing it into the next call to CKFetchDatabaseChangesOperationMBS. If the server returns a CKErrorChangeTokenExpired error, the previousServerChangeToken value was too old and the client should toss its local cache and re-fetch the changes in this record zone starting with a nil previousServerChangeToken.

6.13.12 recordZoneWithIDChanged(zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block that processes a single record zone change.
Notes: The event returns no value and takes the following parameter:

zoneID: The CKRecordZoneID corresponding to the record zone that changed.

6.13.13 recordZoneWithIDWasDeleted(zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event that processes a single record zone deletion.
Notes: The event returns no value and takes the following parameter:

zoneID: The CKRecordZoneID corresponding to the record zone that was deleted.

6.14 class CKFetchNotificationChangesOperationMBS

6.14.1 class CKFetchNotificationChangesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKFetchNotificationChangesOperationMBS object retrieves unread CKNotification objects from a container.
Notes: Notification objects contain the data associated with push notifications that have already been sent to an app. Fetch notification objects to identify changes that your app might have missed or to retrieve the complete push notification payload, which might have been truncated when it was delivered to the device.

After fetching the current notifications, you can mark some or all of them as read using a CKMarkNotificationsReadOperation object. Marking a notification as read prevents it from being returned by subsequent fetch operations.

The events to process the fetched notifications are executed serially on an internal queue that is managed by the operation object. Your events must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

The completion event is called after the operation executes and returns its results to you. You can use a completed event to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion event you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKOperationMBS class.

6.14.2 Methods

6.14.3 Constructor(previousServerChangeToken as CKServerChangeTokenMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object set to fetch notification changes.
Notes: previousServerChangeToken: The change token from a previous fetch operation, or nil to fetch all changes. When you specify a change token, the operation object fetches only the changes that occurred since the change token was generated.

Returns an initialized operation object.

The first time you fetch notifications, specify nil for the previousServerChangeToken to get information about all push notifications that have been sent. When that operation finishes executing, it passes a change token to your completion block. Save that change token and use it to initialize new operation objects that fetch only the newly generated notification objects.

If you mark one or more notifications as read using a CKMarkNotificationsReadOperationMBS object, those notifications are not returned, even if you specify nil for previousServerChangeToken.

6.14.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.14.5 Properties

6.14.6 moreComing as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating that more notifications are waiting to be delivered.
Notes: Access this property only from your fetchNotificationChangesCompletionBlock block. When the value is true, your completed event should create another CKFetchNotificationChangesOperation object and use it to fetch the next batch of results. Use the change token passed to your completion block to configure the new operation object. Specifying the change token prevents the new operation object from returning older notifications.
(Read only property)

6.14.7 previousServerChangeToken as CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The change token from a previous fetch operation.
Notes: Assign the token you received previously from the fetchNotificationChangesCompleted event to this property. Doing so yields only the notifications that have arrived since your last fetch operation. If you specify nil for this parameter, the operation starts fetching notifications at the beginning of the list and returns all but those that have already been marked as read.
(Read and Write property)

6.14.8 resultsLimit as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The maximum number of results to fetch with this operation.
Notes: Use this property to limit the number of push notifications processed in one operation. When the

results limit is reached, the operation object updates the original token you provided and returns it to you to let you know that more results are available.
(Read and Write property)

6.14.9 Events

6.14.10 fetchNotificationChangesCompleted(serverChangeToken as CKServerChangeTokenMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event that is executed after all requested notifications are fetched.
Notes: The event returns no value and takes the following parameters:

serverChangeToken: An opaque object that you can use during subsequent fetch operations to retrieve the next batch of notifications.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this event only once, which represents your last chance to process the operation results. This event is executed after all individual progress events but before the operation,Äôs completion event. The event is executed serially with respect to the other progress events of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

In your event, check the value of the moreComing property to determine if there are more results waiting on the server. If the value of that property is treu, use the value in the serverChangeToken parameter to configure a new CKFetchNotificationChangesOperationMBS object to fetch the next batch of results.

6.14.11 notificationChanged(notification as CKNotificationMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event that processes a single push notification.
Notes: The event returns no value and takes the following parameter:

notification: The CKNotification object corresponding to a push notification. The actual object passed to this method is a concrete subclass of CKNotification that contains specific details about the source of the change.

The operation object executes this event once for each push notification that is found. Each time the event

is executed, it is executed serially with respect to the other progress events of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.15 class CKFetchRecordChangesOperationMBS

6.15.1 class CKFetchRecordChangesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKFetchRecordChangesOperation object reports on the changed and deleted records in the specified record zone.
Notes: Use this type of operation object to optimize fetch operations for locally managed sets of records. Specifically, use it when you maintain a local cache of your record data and need to synchronize that cache periodically with the server.

To get the most benefit out of a CKFetchRecordChangesOperation object, you must maintain a local cache of the records from the specified zone. Each time you fetch changes from that zone, the server provides a token that identifies your request. With each subsequent fetch request, you initialize the operation object with the token from the previous request, and the server returns only the records that changed since that request.

The blocks you assign to process the fetched records are executed serially on an internal queue managed by the operation. Your blocks must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

The completed event is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.

Available in 10.10 and newer and deprecated with 10.12.
Subclass of the CKDatabaseOperationMBS class.

6.15.2 Methods

6.15.3 Constructor(recordZoneID as CKRecordZoneIDMBS, previousServerChangeToken as CKServerChangeTokenMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch changes in the specified record zone.
Notes: recordZoneID: The zone containing the records you want to fetch. The zone can be a custom zone. Syncing the default zone is not supported.
previousServerChangeToken: The change token from a previous fetch operation. This is the token passed to your fetchRecordChangesCompleted event during a previous fetch operation. Use this token to limit the

returned data to only those changes that have occurred since you last made the same fetch request. If you specify nil for this parameter, the operation object fetches all records and their contents.

Returns an initialized operation object.

When initializing the operation object, use the token from a previous fetch request if you have one. You can archive tokens and write them to disk for later use if needed.

The returned operation object is configured to retrieve all changed fields of the record, including any assets stored in those fields. If you want to minimize the amount of data returned even further, configure the desiredKeys property with the subset of keys whose values you want to fetch.

After initializing the operation, associate at least one progress block with the operation object (excluding the completion block) to process the results.

6.15.4 desiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property are included in the returned record. The default value is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and include at least one field name from each record type.

If you intend to specify the desired set of keys, set the value of this property before executing the operation or submitting it to a queue.

6.15.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.15.6 setDesiredKeys(desiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property are included in the returned record. The default value is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and include at least one field name from each record type.

If you intend to specify the desired set of keys, set the value of this property before executing the operation or submitting it to a queue.

6.15.7 Properties

6.15.8 moreComing as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating that more results are available.
Notes: If the server is unable to deliver all of the changed results with this operation object, it sets this property to true before executing the fetchRecordChangesCompleted event. To fetch the remaining changes, create a new CKFetchRecordChangesOperationMBS object using the change token returned by the server. (Read only property)

6.15.9 previousServerChangeToken as CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The token that identifies the starting point for retrieving changes.
Notes: Each fetch request returns a unique token in addition to any changes. The token is passed as a parameter to your fetchRecordChangesCompleted event. During a subsequent fetch request, providing the previous token causes the server to return only the changes that have occurred since the last fetch request. Tokens are opaque data objects that you can write to disk safely and reuse later.

Typically, you set the value of this property when you initialize the operation object. If you intend to change the record zone, update the value of the property before executing the operation or submitting it to a queue. (Read and Write property)

6.15.10 recordZoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the record zone whose records you want to fetch.
Notes: Typically, you set the value of this property when you initialize the operation object. If you intend to change the record zone, update the value before executing the operation or submitting it to a queue.
(Read and Write property)

6.15.11 resultsLimit as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function:

The maximum number of changed records to report with this operation object.
Language
Notes:

Use this property to limit the number of results in situations where you expect the number of changed records might be large. The default value is 0, which causes the server to choose an appropriate number of results to return based on dynamic conditions.

When the number of returned results exceeds the results limit, the operation object sets the moreComing property to true before executing the fetchRecordChangesCompleted event. In your event, check the value of that property, and if it is true, create a new CKFetchRecordChangesOperationMBS object to fetch more results.
(Read and Write property)

6.15.12 Events

6.15.13 fetchRecordChangesCompleted(serverChangeToken as CKServerChangeTokenMBS, clientChangeTokenData as MemoryBlock, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when all changes have been reported.
Notes: The block returns no value and takes the following parameters:

serverChangeToken: The new change token from the server. You can store this token locally and use it during subsequent fetch operations to limit the results to records that changed since this operation executed.
clientChangeToken: The last client change token received from this device. If this change token is not the last change token you provided, the server may not have received the associated changes.

operationError: An error object containing information about a problem, or nil if the changes are retrieved successfully.

Your implementation of this block should check the moreComing property of the operation object to ensure that the server was able to deliver all results. If that property is set to true, you must create another operation object (using the value in the serverChangeToken parameter) to fetch any remaining changes.

The operation object executes this block only once, at the conclusion of the operation. It is executed after all individual change blocks but before the operation,Äôs completed event. The block is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.15.14 recordChanged(record as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the contents of a changed record.
Notes: The event returns no value and takes the following parameters:

record: The record that changed. If you specified a value for the desiredKeys property, the record only contains the fields specified in the desiredKeys property.

The operation object executes this event once for each record in the zone that changed since the previous fetch request. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation. If no records changed, the block is not executed.

If you intend to use this event to process results, set it before executing the operation or submitting it to a queue.

6.15.15 recordWithIDWasDeleted(recordID as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the ID of a record that was deleted.
Notes: The block returns no value and takes the following parameters:

recordID: The ID of the record that was deleted.

The operation object executes this block once for each record in the zone that was deleted since the previous fetch request. Each time the event is executed, it is executed serially with respect to the other progress blocks of the operation. If no records were deleted, this event is not executed.

If you intend to use this event to process results, set it before executing the operation or submitting it to a queue.

6.16 class CKFetchRecordsOperationMBS

6.16.1 class CKFetchRecordsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKFetchRecordsOperation object retrieves CKRecord objects (whose IDs you already know) from iCloud.
Notes: Use this operation to retrieve the entire contents of each record or only a subset of its contained values. As records become available, the operation object reports progress about the state of the operation to several different blocks, which you can use to process the results.

Fetching records is a common use of CloudKit, even if your app does not cache record IDs locally. For example, when you fetch a record related to the current record through a CKReference object, you use the ID in the reference to perform the fetch.

The blocks you assign to process the fetched records are executed serially on an internal queue managed by the fetch records operation. Your blocks must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

In addition to data records, a fetch records operation can fetch the current user record. The fetchCurrentUserRecordOperation() method returns a specially configured operation object that retrieves the current user record. That record is a standard CKRecord object whose contents are empty initially. You can add data to the user record and save it as needed. Because a discoverable user record can be accessed by other users of the app, never store sensitive personal information such as passwords in the user record. If you must store sensitive information about a user, do so in a separate record that is accessible only to that user.

The completed event is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKDatabaseOperationMBS class.

6.16.2 Methods

6.16.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

6.16.4 Constructor(recordIDs() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch the records with the specified IDs.
Notes: recordIDs: An array of CKRecordIDMBS objects representing the records you want to retrieve. This parameter is used to initialize the value in the recordIDs property. If specify nil, you must assign an appropriate value to the recordIDs property before executing the operation.

If any of the objects in the array are not CKRecordIDMBS objects, this method raises an exception.

Returns an initialized operation object.

The returned operation object is configured to retrieve all fields of the record, including any assets stored in those fields. If you want to minimize the amount of data returned initially, configure the desiredKeys property with the subset of keys whose values you want to retrieve.
See also:

- 6.16.3 Constructor 277

6.16.5 desiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property are included in the returned record. The default value of this property is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and should include at least one field name from each record type.

If you intend to specify a value other than nil, do so before executing the operation or submitting the operation object to a queue.

6.16.6 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.16.7 fetchCurrentUserRecordOperation as CKFetchRecordsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an operation object that can be used to fetch the current user record.
Notes: The returned operation object searches for the single record corresponding to the current user record. You must associate at least one progress block with the operation object (excluding the completed event) to process the results.

6.16.8 recordIDs as CKRecordIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The array of IDs corresponding to the records to fetch.
Notes: Use this property to view or change the IDs of the records you want to retrieve. Each item in the array must be a CKRecordIDMBS object. If you used the fetchCurrentUserRecordOperation method to create the operation object, the contents of this property are ignored and the value is set to nil.

If you intend to specify a value other than nil, do so before executing the operation or submitting the operation object to a queue. The records you fetch do not need to be in the same record zone. The record ID for each record provides the zone information needed by the server to fetch the corresponding record.

6.16.9 setDesiredKeys(desiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Set the fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property are included in the returned record. The default value of this property is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and should include at least one field name from each record type.

If you intend to specify a value other than nil, do so before executing the operation or submitting the operation object to a queue.

6.16.10 setRecordIDs(IDs() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the array of IDs corresponding to the records to fetch.
Notes: Use this property to view or change the IDs of the records you want to retrieve. Each item in the array must be a CKRecordIDMBS object. If you used the fetchCurrentUserRecordOperation method to create the operation object, the contents of this property are ignored and the value is set to nil.

If you intend to specify a value other than nil, do so before executing the operation or submitting the operation object to a queue. The records you fetch do not need to be in the same record zone. The record ID for each record provides the zone information needed by the server to fetch the corresponding record.

6.16.11 Events

6.16.12 fetchRecordsCompleted(recordsByRecordID as Dictionary, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute after all records are fetched or have received appropriate errors.
Notes: The event returns no value and takes the following parameters:

recordsByRecordID: A dictionary containing the records that are retrieved successfully. Each key in the dictionary is a CKRecordID object corresponding to a record you requested. The value of each key is the corresponding CKRecord object that was retrieved from the database.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this event only once and is your last chance to process the operation results. The event is executed after all of the individual progress events but before the operation,Äôs completed event. The event is executed serially with respect to the other progress events of the operation.

This event reports an error of type CKErrorPartialFailure when it retrieves only some of the records successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is an NSDictionary object. The keys of that dictionary are the IDs of the records that were not retrieved, and the corresponding values are error objects containing information about what happened.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.16.13 RecordCompleted(record as CKRecordMBS, recordID as CKRecordIDMBS, error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the results of a single record are available.
Notes: The event returns no value and takes the following parameters:

record: The retrieved record, or nil if the specified record cannot be retrieved.
recordID: The ID of the record. This value corresponds to one of the IDs you specified in the recordIDs property.
error: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this block once for each record ID in the recordIDs property. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.16.14 RecordProgress(recordID as CKRecordIDMBS, progress as Double)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with progress information for individual records.
Notes: The event returns no value and takes the following parameters:

recordID: The ID of the record that is being retrieved.
progress: The amount of the record that has been downloaded, represented as a percentage of the total. The range of this value is 0.0 to 1.0, where 0.0 means nothing has been downloaded, and 1.0 means the download is complete.

The operation object executes this block zero or more times for each record ID in the recordIDs property. Each time the block is executed, it is executed serially with respect to the other progress events of the operation. You can use this event to track the ongoing progress of the download operation and possibly to provide feedback to the user.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

6.17 class CKFetchRecordZoneChangesOperationMBS

6.17.1 class CKFetchRecordZoneChangesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that fetches record changes across the given record zone.
Notes: Subclass of the CKDatabaseOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 21.2pr7
- MBS Xojo Plugins, version 21.2pr5

6.17.2 Methods

6.17.3 Constructor(recordZoneIDs() as CKRecordZoneIDMBS, optionsByRecordZoneID as Dictionary = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch record zone changes.
Notes: The returned operation object is configured to retrieve all record zones passed in. If you want to minimize the amount of data returned even further, configure the CKFetchRecordZoneChangesOptionsMBS property for each record zone.

After initializing the operation, associate at least one progress block with the operation object (excluding the completion block) to process the results.

optionsByRecordZoneID: The options per zone.
recordZoneIDs: The record zones that should be fetched.

6.17.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.17.5 recordZoneIDs as CKRecordZoneIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the record zones whose records you want to fetch.
Notes: Typically, you set the value of this property when you initialize the operation object. If you intend to change the record zones, update the value before executing the operation or submitting it to a queue.

6.17.6 setRecordZoneIDs(IDs() as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the record zones whose records you want to fetch.
Notes: Typically, you set the value of this property when you initialize the operation object. If you intend to change the record zones, update the value before executing the operation or submitting it to a queue.

6.17.7 Properties

6.17.8 fetchAllChanges as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether repeated requests should be sent to the server.
Notes: If true, this operation sends repeated requests to the server until CKFetchRecordZoneChangesOperationMBS is false. The server calls recordZoneFetchCompletionBlock with an incremental change token after every request. The default value is true.
(Read and Write property)

6.17.9 optionsByRecordZoneID as Dictionary

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Options for each zone that is retrieved from the server.
Notes: Each CKRecordZoneIDMBS can have its own CKFetchRecordZoneChangesOptionsMBS object that allows you to configure what is fetched for that zone. See CKFetchRecordZoneChangesOptionsMBS for details on what options can be set.
(Read and Write property)

6.17.10 Events

6.17.11 fetchRecordZoneChangesCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to use to process the record zone changes.
Notes: The event has no return value and takes the following parameter:

operationError: An error object containing information about a problem, or nil if the record zone changes were retrieved successfully.

6.17.12 recordChanged(record as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the contents of a changed record.
Notes: The block returns no value and takes the following parameter:

record: The record that changed. If you specified a value for the desiredKeys property, the record only contains the fields specified in the desiredKeys property.

The operation object executes this block once for each record in the zone that changed since the previous fetch request. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation. If no records changed, the block is not executed.

If you intend to use this block to process results, set it before executing the operation or submitting it to a queue.

6.17.13 recordWithIDWasDeleted(recordID as CKRecordIDMBS, recordType as string)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the ID of a record that was deleted.
Notes: The event returns no value and takes the following parameter:

recordID: The ID of the record that was deleted.

The operation object executes this block once for each record that was deleted since the previous fetch request. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation. If no records were deleted, this block is not executed.

If you intend to use this block to process results, set it before executing the operation or submitting it to a queue.

6.17.14 recordZoneChangeTokensUpdated(recordZoneID as CKRecordZoneIDMBS, serverChangeToken as CKServerChangeTokenMBS, clientChangeTokenData as MemoryBlock)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the change token has been updated.
Notes: The event returns no value and takes the following parameters:

recordZoneID: The ID of the zone with the updated token.
serverChangeToken: The new change token from the server. You can store this token locally and use it during subsequent fetch operations to limit the results to records that changed since this operation executed.
clientChangeTokenData: The last client change token received from this device. If this change token is not the last change token you provided, the server may not have received the associated changes.

The operation object executes this block once for each record zone. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this block to process results, set it before executing the operation or submitting it to a queue.

6.17.15 recordZoneFetchCompleted(recordZoneID as CKRecordZoneIDMBS, serverChangeToken as CKServerChangeTokenMBS, clientChangeTokenData as MemoryBlock, moreComing as boolean, recordZoneError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the fetch for a zone has completed.
Notes: The block returns no value and takes the following parameters:

recordZoneID: The ID of the zone with the updated token.
serverChangeToken: The current server change token to be stored and used in subsequent CKFetchRecordZoneChangesOperation instances.
clientChangeTokenData: The last client change token received from this device. If this change token is not the last change token you provided, the server may not have received the associated changes.
moreComing: Indicates if this is the last record zone change. If fetchAllChanges is false, it is the responsibility of the client to create additional CKFetchRecordZoneChangesOperationMBS instances for the additional changes.
recordZoneError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The client is responsible for saving the change token at the end of the operation and passing it into the next

call to CKFetchRecordZoneChangesOperationMBS. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting it to a queue.

6.18 class CKFetchRecordZoneChangesOptionsMBS

6.18.1 class CKFetchRecordZoneChangesOptionsMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Options when fetching record zone changes.

6.18.2 Methods

6.18.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.18.4 desiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property are included in the returned record. The default value is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and include at least one field name from each record type.

If you intend to specify the desired set of keys, set the value of this property before executing the operation or submitting it to a queue.

6.18.5 setDesiredKeys(desiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the fields to fetch for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the fetch operation. This property contains an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the keys in this property

are included in the returned record. The default value is nil, which causes all keys of the record to be fetched.

Because the records you fetch can be of different types, the array should contain the merged set of all field names for the requested records and include at least one field name from each record type.

If you intend to specify the desired set of keys, set the value of this property before executing the operation or submitting it to a queue.

6.18.6 Properties

6.18.7 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.18.8 previousServerChangeToken as CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The token that identifies the starting point for retrieving changes.
Notes: Each fetch request returns a unique token in addition to any changes. The token is passed as a parameter to your fetchRecordChangesCompleted event. During a subsequent fetch request, providing the previous token causes the server to return only the changes that have occurred since the last fetch request. Tokens are opaque data objects that you can write to disk safely and reuse later.

Typically, you set the value of this property when you initialize the operation object. If you intend to change the record zone, update the value of the property before executing the operation or submitting it to a queue. (Read and Write property)

6.18.9 resultsLimit as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The maximum number of changed records to report with this operation object.
Notes: Use this property to limit the number of results in situations where you expect the number of changed records might be large. The default value is 0, which causes the server to choose an appropriate number of results to return based on dynamic conditions.

When the number of returned results exceeds the results limit, the operation object sets the moreComing property to true before executing the fetchRecordChangesCompleted event. In your block, check the value of that property, and if it is true, create a new CKFetchRecordChangesOperationMBS object to fetch more results.
(Read and Write property)

6.19 class CKFetchRecordZonesOperationMBS

6.19.1 class CKFetchRecordZonesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKFetchRecordZonesOperation object retrieves CKRecordZone objects (whose IDs you already know) from iCloud.
Notes: Use this operation object to fetch record zones so that you can ascertain their capabilities.

The completed event is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKDatabaseOperationMBS class.

6.19.2 Methods

6.19.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The default constructor.
See also:

- 6.19.4 Constructor(recordZoneIDs() as CKRecordZoneIDMBS) 290

6.19.4 Constructor(recordZoneIDs() as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch the specified record zones.
Notes: zoneIDs: An array of CKRecordZoneID objects representing the zones you want to retrieve. This parameter is used to initialize the value in the recordZoneIDs property. If you specify nil, you must assign a value to the recordZoneIDs property before executing the operation.

Returns an initialized operation object.
See also:

- 6.19.3 Constructor 290

6.19.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.19.6 fetchAllRecordZonesOperation as CKFetchRecordZonesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an operation object that can be used to fetch all record zones in the current database.

6.19.7 recordZoneIDs as CKRecordZoneIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the record zones you want to retrieve.
Notes: Use this property to view or change the IDs of the record zones you want to retrieve. Each item in the array must be a CKRecordZoneID object. If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

If you used the fetchAllRecordZonesOperationMBS method to create the operation object, the contents of this property are ignored and the default value is set to nil.

6.19.8 setRecordZoneIDs(IDs() as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the record zones you want to retrieve.
Notes: Use this property to view or change the IDs of the record zones you want to retrieve. Each item in the array must be a CKRecordZoneID object. If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

If you used the fetchAllRecordZonesOperationMBS method to create the operation object, the contents of this property are ignored and the default value is set to nil.

6.19.9 Events

6.19.10 fetchRecordZonesCompleted(recordZonesByZoneID as Dictionary, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the fetch results.
Notes: The event returns no value and takes the following parameters:

recordZonesByZoneID: A dictionary that maps the requested IDs to the retrieved objects. The keys in the dictionary are the CKRecordZoneID objects you requested, and the values are the corresponding CKRecordZone objects.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this block only once and it is your only chance to process the operation results. The block is executed before the operation,Äôs completion block. Your block must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

This block reports an error of type CKErrorPartialFailure when it retrieves only some of the record zones successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the record zones that were not retrieved, and the corresponding values are error objects containing information about what happened.

6.20 class CKFetchShareMetadataOperationMBS

6.20.1 class CKFetchShareMetadataOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that fetches shared record metadata for one or more shares.
Notes: This operation may be run in any container that the client has access to.
Subclass of the CKOperationMBS class.

6.20.2 Methods

6.20.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.20.4 Constructor(URLs() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch the metadata for the specified shares.
Notes: shareURLs: An array of URLs. This parameter is used to initialize the value in the shareURLs property. If you specify nil, you must assign an appropriate value to the shareURLs property before executing the operation.

Returns an initialized operation object.

6.20.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.20.6 rootRecordDesiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Keys to be fetched if the root record is to be fetched.
Notes: This property declares which user-defined keys should be fetched and added to the resulting root record. This property is only consulted if shouldFetchRootRecord is set to true.

If this property is set to nil, the entire root record is downloaded.
If this property is set to an empty array, no user fields are downloaded.

This property defaults to nil.

6.20.7 setRootRecordDesiredKeys(rootRecordDesiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the keys to be fetched if the root record is to be fetched.
Notes: This property declares which user-defined keys should be fetched and added to the resulting root record. This property is only consulted if shouldFetchRootRecord is set to true.

If this property is set to nil, the entire root record is downloaded.
If this property is set to an empty array, no user fields are downloaded.

This property defaults to nil.

6.20.8 setShareURLs(URLs() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the URLs of the shares that you want to process.
Notes: Use this property to view or change the URLs of the share objects that you want to fetch. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.20.9 shareURLs as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The URLs of the shares that you want to process.
Notes: Use this property to view or change the URLs of the share objects that you want to fetch. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.20.10 Properties

6.20.11 shouldFetchRootRecord as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A flag to indicate whether the root record should be retrieved.
Notes: If this property is set to true, the resulting CKShareMetadataMBS has a rootRecord object filled out.
The resulting CKShareMetadata has a rootRecordID property regardless of the value of this property.
This property defaults to false.
(Read and Write property)

6.20.12 Events

6.20.13 fetchShareMetadataCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the operation has completed.
Notes: This block is called when the operation completes. The Complete of the underlying NSOperation is also called if both are implemented.

The event is executed serially with respect to the other progress events of the operation. If you intend to use this event to process results, update the value of this property before executing the operation or submitting the operation object to a queue.

6.20.14 ShareMetadataFetched(shareURL as String, shareMetadata as CKShareMetadataMBS, error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each share metadata that has been fetched from the server.
Notes: If error is nil then the share was successfully accepted. Each time the event is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.21 class CKFetchShareParticipantsOperationMBS

6.21.1 class CKFetchShareParticipantsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An operation that fetches the participants of a shared record.
Notes: Subclass of the CKOperationMBS class.

6.21.2 Methods

6.21.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch share participants.
Notes: You can use the returned CKFetchShareParticipantsOperation only once. When executed, this query object fetches the share participants.

Note
If userIdentityLookupInfos is not set prior to executing the operation, it returns immediately with no results.
See also:

- 6.21.4 Constructor(userIdentityLookupInfos() as CKUserIdentityLookupInfoMBS) 296

6.21.4 Constructor(userIdentityLookupInfos() as CKUserIdentityLookupInfoMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch the specified share participants.
Notes: userIdentityLookupInfos: An array of CKUserIdentityLookupInfoMBS objects. This parameter is used to initialize the value in the userIdentityLookupInfos property. If you specify nil, you must assign an appropriate value to the userIdentityLookupInfosuserIdentityLookupInfos property before executing this operation.

Returns an initialized operation object.
See also:

- 6.21.3 Constructor 296

6.21.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.21.6 setUserIdentityLookupInfos(userIdentityLookupInfos() as CKUserIdentityLookupInfoMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the user identities that are used to fetch the share participants.
Notes: Use this property to view or change the user identities you want to fetch against. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

Note
If userIdentityLookupInfos is not set prior to executing the operation, it returns immediately with no results.

6.21.7 userIdentityLookupInfos as CKUserIdentityLookupInfoMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The user identities that are used to fetch the share participants.
Notes: Use this property to view or change the user identities you want to fetch against. If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.

Note
If userIdentityLookupInfos is not set prior to executing the operation, it returns immediately with no results.

6.21.8 Events

6.21.9 fetchShareParticipantsCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the operation has completed.
Notes: This event is called when the operation complete. The Comepleted of the underlying NSOperation is also called if both are set.

The event is executed serially with respect to the other progress blocks of the operation. If you intend to use this event to process results, update the value of this property before executing the operation or submitting the operation object to the queue.

6.21.10 shareParticipantFetched(participant as CKShareParticipantMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each share participant that has been fetched from the server.
Notes: Each time the event is executed, it is executed serially with respect to the other progress events of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.22 class CKFetchSubscriptionsOperationMBS

6.22.1 class CKFetchSubscriptionsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKFetchSubscriptionsOperation object retrieves CKSubscription objects (whose IDs you already know) from iCloud and can fetch all subscriptions associated with the current user.
Notes: You might fetch subscription objects so you can examine or modify their parameters; for example, to adjust the delivery options for push notifications generated by the subscription.

The completion event is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKDatabaseOperationMBS class.

6.22.2 Methods

6.22.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.22.4 Constructor(subscriptionIDs() as String) 299

6.22.4 Constructor(subscriptionIDs() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to fetch the specified subscriptions.
Notes: subscriptionIDs: An array of strings, each of which contains the ID of a subscription object you want to retrieve. This parameter is used to initialize the value in the subscriptionIDs property. If you specify nil, you must assign an appropriate value to the subscriptionIDs property before executing the operation.

Returns an initialized operation object.
See also:

- 6.22.3 Constructor 299

6.22.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.22.6 fetchAllSubscriptionsOperation as CKFetchSubscriptionsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an operation object that retrieves all of the user,Äôs subscription objects in the current database.
Notes: Returns a newly allocated operation object that fetches the active subscription objects operating on behalf of the current user.

6.22.7 setSubscriptionIDs(emails() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the subscription objects you want to retrieve.
Notes: Use this property to view or change the IDs of the subscription objects you want to retrieve. Each item in the array must be a string whose value is the ID of the subscription object you want to retrieve. If you intend to specify or change the value for this property, do so before executing the operation or submitting the operation object to a queue.

If you used the fetchAllSubscriptionsOperation method to create the operation object, the contents of this property are ignored and its value is set to nil.

6.22.8 subscriptionIDs as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the subscription objects you want to retrieve.
Notes: Use this property to view or change the IDs of the subscription objects you want to retrieve. Each item in the array must be a string whose value is the ID of the subscription object you want to retrieve. If you intend to specify or change the value for this property, do so before executing the operation or submitting the operation object to a queue.

If you used the fetchAllSubscriptionsOperation method to create the operation object, the contents of this property are ignored and its value is set to nil.

6.22.9 Events

6.22.10 fetchSubscriptionCompleted(subscriptionsBySubscriptionID as Dictionary, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the fetch results.
Notes: The event returns no value and takes the following parameters:

subscriptionsBySubscriptionID: A dictionary that maps the requested IDs to the retrieved objects. The keys in the dictionary are the NSString objects with the IDs of the subscription objects you requested and the values are the corresponding CKSubscription objects.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this block only once and it is your only chance to process the operation results. The block is executed before the operation,Äôs completion block. Your block must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

This block reports an error of type CKErrorPartialFailure when it retrieves only some of the subscriptions successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the subscriptions that were not retrieved and the corresponding values are error objects containing information about what happened.

6.23 class CKFetchWebAuthTokenOperationMBS

6.23.1 class CKFetchWebAuthTokenOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The CKFetchWebAuthTokenOperation object fetches a web authentication token given an API token that you obtain from CloudKit Dashboard.
Notes: Use the add method in the CKDatabase class to add this operation to the operation queue of the public database.
Subclass of the CKDatabaseOperationMBS class.

6.23.2 Methods

6.23.3 Constructor(APIToken as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a CKFetchWebAuthTokenOperationMBS object with the specified API token.
Notes: APIToken: An API token that allows access to an app,Äôs container.

6.23.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.23.5 Properties

6.23.6 APIToken as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An API token that allows access to an app,Äôs container.
Notes: (Read and Write property)

6.23.7 Events

6.23.8 fetchWebAuthTokenCompleted(webAuthToken as string, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when this CKFetchWebAuthTokenOperation object fetches the web authentication token.
Notes: The event returns no value and takes the following parameters:

webAuthToken: If the operation is successful, the web authentication token; otherwise, nil.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

The operation object executes this block only once and it is your only chance to process the operation results. Your block must be capable of executing on a background thread, so any tasks that require access to the main thread must be redirected accordingly.

6.24 class CKLocationSortDescriptorMBS

6.24.1 class CKLocationSortDescriptorMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKLocationSortDescriptor object sorts records containing location data based on their distance from a location that you specify.
Notes: You can add a location sort descriptor to your queries when searching for records. At creation time, you must provide the sort descriptor with a key whose value is a CLLocation object. The sort descriptor uses the value of that key to perform the sort.

Distances are computed by drawing a direct line between the two locations that follows the curvature of the Earth. Distances do not take into account altitude changes between the two locations.
Subclass of the NSSortDescriptorMBS class.

6.24.2 Methods

6.24.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.24.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Private constructor.
See also:

- 6.24.5 Constructor(key as string, relativeLocation as Variant) 304

6.24.5 Constructor(key as string, relativeLocation as Variant)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a location sort descriptor object.
Notes: key: The name of the key whose value is a CLLocationMBS object. The key must belong to the records being sorted. The sort descriptor uses this key to retrieve the corresponding value from the record.
relativeLocation: The reference location to use when sorting. Records are sorted based on their distance to

this location.

Returns an initialized location sort descriptor object, or nil if the object cannot be initialized.

During sorting, the sort descriptor computes the distance between the value in the relativeLocation parameter and the location value found in the specified key of each record. It then sorts the records in ascending order using the distance between the two points. You cannot change the sort order.

relativeLocation must be a CLLocationMBS object.
See also:

- 6.24.4 Constructor 304

6.24.6 Properties

6.24.7 relativeLocation as Variant

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The reference location against which records are sorted.
Notes: Value is a CLLocationMBS.
(Read only property)

6.25 class CKMarkNotificationsReadOperationMBS

6.25.1 class CKMarkNotificationsReadOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKMarkNotificationsReadOperation object marks push notifications as read by your app so that they do not show up in future fetch results.
Notes: If your app uses push notifications to track changes to records, you can use this operation object to note which push notifications do not need to be processed again.

The completion event is called after the operation executes and returns its results to you. You can use a completion block to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKOperationMBS class.

6.25.2 Methods

6.25.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.25.4 Constructor(IDs() as CKNotificationIDMBS) 306

6.25.4 Constructor(IDs() as CKNotificationIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to mark the specified notifications as read.
Notes: notificationIDs: An array of CKNotificationIDMBS objects representing the notifications you want to mark as read. Use this parameter to initialize the value in the notificationIDs property. If you specify nil, assign an appropriate value to the notificationIDs property before executing the operation.

If any objects in the array are not CKNotificationIDMBS objects, this method raises an exception.

Returns an initialized operation object.
See also:

- 6.25.3 Constructor 306

6.25.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.25.6 notificationIDs as CKNotificationIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The array of notifications to mark as read.
Notes: Use this property to view or change the IDs of the notifications you want to mark as read. Each item in the array must be a CKNotificationIDMBS object. If you intend to specify a value for this property, do so before executing the operation or submitting the operation object to a queue.

6.25.7 setNotificationIDs(IDs() as CKNotificationIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the array of notifications to mark as read.
Notes: Use this property to view or change the IDs of the notifications you want to mark as read. Each item in the array must be a CKNotificationIDMBS object. If you intend to specify a value for this property, do so before executing the operation or submitting the operation object to a queue.

6.25.8 Events

6.25.9 markNotificationsReadCompleted(notificationIDsMarkedRead() as CKNotificationIDMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when all results of the operation are known.
Notes: The event returns no value and takes the following parameters:

notificationIDsMarkedRead: An array of CKNotificationID objects corresponding to the notifications that were successfully marked as read.
operationError: An error object containing information about a problem, or nil if all notifications are updated successfully.

The operation object executes this block only once and it is your only chance to process the operation results. If you intend to use this block to process results, set it before executing the operation or submitting the

operation object to a queue.

This block reports an error of type CKErrorPartialFailure when it is able to update only some of the notifications successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the notifications that were not updated, and the corresponding values are error objects containing information about what happened.

6.26 class CKModifyBadgeOperationMBS

6.26.1 class CKModifyBadgeOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKModifyBadgeOperation object updates the badge value applied to the app,Äôs icon.
Notes: This operation object can update the badge for the current device or for all of the user,Äôs devices.

The completion event is called after the operation executes and returns its results to you. You can use a completion event to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion event you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKOperationMBS class.

6.26.2 Methods

6.26.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.26.4 Constructor(badgeValue as Integer) 309

6.26.4 Constructor(badgeValue as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object for setting the badge of the app.
Notes: badgeValue: The numerical value of the app,Äôs badge. Specify 0 to remove the badge.
See also:

- 6.26.3 Constructor 309

6.26.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.26.6 Properties

6.26.7 badgeValue as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The numerical value to apply to the app,Äôs badge.
Notes: The initial value of this property is set to the value you specified using the Constructor. If you intend to change the value, do so before executing the operation or submitting it to a queue.
(Read and Write property)

6.26.8 Events

6.26.9 modifyBadgeCompleted(operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with the results of the operation.
Notes: The event returns no value and takes the following parameters:

operationError: An error object containing information about a problem, or nil if the badge is applied successfully.

This event is executed only once and represents your only chance to process the operation results. If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue.

6.27 class CKModifyRecordsOperationMBS

6.27.1 class CKModifyRecordsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKModifyRecordsOperationMBS object saves changes to one or more CKRecordMBS objects.
Notes: After modifying the fields of a record, use this type of operation object to save those changes to a database. You also use instances of this class to delete records permanently from a database.

see
https://developer.apple.com/reference/cloudkit/ckmodifyrecordsoperation
Subclass of the CKDatabaseOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 21.2pr8

6.27.2 Methods

6.27.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.27.4 Constructor(recordsToSave() as CKRecordMBS, recordIDsToDelete() as CKRecordIDMBS) 311

6.27.4 Constructor(recordsToSave() as CKRecordMBS, recordIDsToDelete() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to save and delete the specified records.
Notes: recordsToSave: An array of CKRecordMBS objects representing the records to save, if any. You may specify nil for this parameter.
recordIDsToDelete: An array of CKRecordIDMBS objects representing the records you want to delete, if any. You may specify nil for this parameter.

Returns an initialized operation object.

The records you intend to save or delete must all reside in the same database, which you can specify when you configure the operation object. Saving a record that is not in the current database creates it in the database. Trying to delete a record that does not exist in the current database returns an error for that

record.
See also:

- 6.27.3 Constructor 311

6.27.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.27.6 recordIDsToDelete as CKRecordIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the records to delete permanently from the database.
Notes: This property contains the array of CKRecordID objects that identify the records to delete. The initial contents of the array are set to the records you specified in the Constructor.

If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

When deleting records, the operation object reports progress only on the records you specify in this property. Deleting records can trigger the deletion of related records if there is an owner-owned relationship between the records involving a CKReference object. When additional deletions occur, they are not reported to the progress blocks of this object. For that reason, it is important to understand the implications of the ownership model you use when you relate records to each other through a CKReference object. For more information about owner-owned relationships, see CKReference.

6.27.7 recordsToSave as CKRecordMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The records to save to the database.
Notes: This property contains the array of CKRecordMBS objects that you want to save. The initial contents of the array are set to the records you specified in the constructor. You can modify this array as needed before executing the operation. The records must all target the same database but may belong to different zones in the database.

If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.27.8 setrecordIDsToDelete(IDs() as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the records to delete permanently from the database.
Notes: This property contains the array of CKRecordID objects that identify the records to delete. The initial contents of the array are set to the records you specified in the Constructor.

If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

When deleting records, the operation object reports progress only on the records you specify in this property. Deleting records can trigger the deletion of related records if there is an owner-owned relationship between the records involving a CKReference object. When additional deletions occur, they are not reported to the progress blocks of this object. For that reason, it is important to understand the implications of the ownership model you use when you relate records to each other through a CKReference object. For more information about owner-owned relationships, see CKReference.

6.27.9 setRecordsToSave(IDs() as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the records to save to the database.
Notes: This property contains the array of CKRecordMBS objects that you want to save. The initial contents of the array are set to the records you specified in the constructor. You can modify this array as needed before executing the operation. The records must all target the same database but may belong to different zones in the database.

If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.

6.27.10 Properties

6.27.11 atomic as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the entire operation fails when one or more records in the same zone cannot be written.
Notes: Modifying records atomically prevents you from updating your data in a way that would leave it in an inconsistent state. You use atomic updates when you want to write multiple records to the same record zone. If there is a failure to modify any of the records in a particular zone, no changes are made to the other records in that same zone. The zone itself must have the CKRecordZoneCapabilityAtomic capability for this

behavior to apply. If a record zone does not support the atomic capability, setting this property has no effect.

The default value of this property is true, which causes all modifications within a single record zone to be made atomically. If your operation object contains records in multiple record zones, a failure in one zone does not prevent modifications to records in a different zone. Changing the value of this property to false causes the records to be modified individually, regardless of whether the record zone supports atomic modifications.
(Read and Write property)

6.27.12 clientChangeTokenData as MemoryBlock

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A data token used to track client-side changes to records.
Notes: When you modify records from a fetch operation, specify a client-generated data token using this property to indicate which version of the record you last modified. Compare the data token you supplied to the data token in the next record fetch to confirm the server has successfully received the device,Äôs last modify request.

The default value is nil.

If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.
(Read and Write property)

6.27.13 savePolicy as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The policy to apply when the server contains a newer version of a specific record.
Notes: Each record has a change tag that allows the server to track when that record was saved. When you save a record, CloudKit compares the change tag in your local copy of the record with the one on the server. If the two tags do not match—meaning that there is a potential conflict—the server uses the value in this property to determine how to proceed.

The default value of this property is CKRecordSaveIfServerRecordUnchanged. If you intend to change the value of this property, do so before executing the operation or submitting the operation object to a queue.
(Read and Write property)

6.27.14 Events

6.27.15 modifyRecordsCompleted(savedRecords() as CKRecordMBS, deletedRecordIDs() as CKRecordIDMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute after the status of all changes is known.
Notes: The event returns no value and takes the following parameters:

savedRecords: The array of CKRecordMBS objects you tried to save.
deletedRecordIDs: The array of CKRecordIDMBS objects corresponding to the records you tried to delete.
operationError: An error object containing information about a problem, or nil if the results are saved successfully.

This event is executed only once and represents your last chance to process the operation results. It is executed after all individual progress blocks have completed but before the operation,Äôs completion block. The block is executed serially with respect to the other progress blocks of the operation.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

This block reports an error of type CKErrorPartialFailure when it saves or deletes only some of the records successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the records that were not saved or deleted, and the corresponding values are error objects containing information about what happened.

6.27.16 RecordCompleted(record as CKRecordMBS, error as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the save results of a single record are known.
Notes: The event returns no value and takes the following parameters:

record: A CKRecordMBS object that you attempted to save.
error: An error object containing information about a problem, or nil if the record was saved successfully.

This block is executed once for each record in the recordsToSave property. Each time the block is executed, it is executed serially with respect to the other progress events of the operation.

If you intend to use this event to process results, set it before executing the operation or submitting the operation object to a queue. Use this block to take any actions after the status of saving the record is known.

6.27.17 RecordProgress(record as CKRecordMBS, progress as Double)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute with progress information for an individual record.
Notes: The block returns no value and takes the following parameters:

record: The CKRecordMBS object that is in the process of being saved.
progress: The amount of progress toward saving the record, expressed as a percentage of its total size. This value is a number between 0.0 and 1.0, where 0.0 means none of the record is saved and 1.0 means the entire record has been saved.

The operation object executes this block zero or more times for each record in the recordsToSave property. Each time the block is executed, it is executed serially with respect to the other progress blocks of the operation.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue. Use this block to track the ongoing progress of the upload operation and possibly to provide feedback to the user.

6.27.18 Constants

Constants

Constant	Value	Description
SaveAllKeys	2	One of the constants indicating the policy to apply when saving records. A policy that saves all keys of the record (including those that have not changed) to the server, overwriting any values currently on the server. Keys present only on the server remain unchanged.
SaveChangedKeys	1	One of the constants indicating the policy to apply when saving records. A policy that saves only those fields of the record that actually changed, overwriting any values currently on the server. Unmodified fields are left untouched.
SaveIfServerRecordUnchanged	0	One of the constants indicating the policy to apply when saving records. A policy that saves the record only if the local copy of the record is based on the record still on the server. The server maintains a change tag for each record automatically. When you fetch a record, that change tag is included with the rest of the record,Äôs data. If the change tag in your local record matches the change tag of the record on the server, the save operation proceeds normally. If the server record contains a newer change tag, the record is not saved and an error of type CKErrorServerRecordChanged is reported.

6.28 class CKModifyRecordZonesOperationMBS

6.28.1 class CKModifyRecordZonesOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKModifyRecordZonesOperationMBS object saves or deletes record zones.
Notes: After creating one or more record zones, use this operation object to save those zones to the database. You can also use it to delete existing zones and the records they contain.

The completion event is called after the operation executes and returns its results to you. You can use a completion event to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion block you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.
Subclass of the CKDatabaseOperationMBS class.

6.28.2 Methods

6.28.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 6.28.4 Constructor(recordZonesToSave() as CKRecordZoneMBS, recordZoneIDsToDelete() as CKRecordZoneIDMBS) 317

6.28.4 Constructor(recordZonesToSave() as CKRecordZoneMBS, recordZoneIDsToDelete() as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to save and delete the specified record zones.
Notes: recordZonesToSave: An array of CKRecordZoneMBS objects representing the record zones to save. You may specify nil for this parameter.
recordZoneIDsToDelete: An array of CKRecordZoneIDMBS objects representing the record zones you want to delete, if any. You may specify nil for this parameter.

Returns an initialized operation object.

The record zones you intend to save or delete must all reside in the same database, which you specify when

configuring the operation object. Deleting a record zone also deletes any records it contains. Trying to delete a record zone that does not exist in the current database returns an error.
See also:

- 6.28.3 Constructor 317

6.28.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.28.6 recordZoneIDsToDelete as CKRecordZoneIDMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the record zones to delete permanently from the database.
Notes: This property contains an array of CKRecordZoneIDMBS objects identifying the zones you want to delete. You set the initial contents of this property with the Constructor. You can assign a new array as needed before executing the operation. The record zones must all be located in the same database. You may specify nil or an empty array for this property.

If you intend to change the value of this property, do so before executing the operation or submitting it to a queue.

6.28.7 recordZonesToSave as CKRecordZoneMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record zones to save to the database.
Notes: This property contains an array of CKRecordZoneMBS objects representing the zones you want to save. You set the initial contents of this property with the Constructor. You can assign a new array as needed before executing the operation. The record zones must all target the same database. You may specify nil or an empty array for this property.

If you intend to change the value of this property, do so before executing the operation or submitting it to a queue.

6.28.8 setRecordZoneIDsToDelete(IDs() as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Set the IDs of the record zones to delete permanently from the database.

6.28.9 setRecordZonesToSave(IDs() as CKRecordZoneMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the record zones to save to the database.

6.28.10 Events

6.28.11 modifyRecordZonesCompleted(savedRecordZones() as CKRecordZoneMBS, deletedRecordZoneIDs() as CKRecordZoneIDMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute after the status of all changes is known.
Notes: The event returns no value and takes the following parameters:

savedRecordZones: An array of CKRecordZone objects you tried to save.
deletedRecordZoneIDs: An array of CKRecordZoneID objects corresponding to the record zones you tried to delete.
operationError: An error object containing information about a problem, or nil if the results are saved successfully.

This block is executed only once and represents your only chance to process the operation results. If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

This event reports an error of type CKErrorPartialFailure when it saves or deletes only some of the record zones successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the record zones that were not saved or deleted, and the corresponding values are error objects containing information about what happened.

6.29 class CKModifySubscriptionsOperationMBS

6.29.1 class CKModifySubscriptionsOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKModifySubscriptionsOperation object saves changes to one or more CKSubscription objects.
Notes: After creating or changing the configuration of a subscription object, use this type of operation object to save those changes to an iCloud container. You also use instances of this class to delete subscriptions permanently from a database.

The complete event is called after the operation executes and returns its results to you. You can use a completion event to perform housekeeping chores related to the operation, but do not use it to process the results of the operation itself. Any completion event you specify should be prepared to handle the failure of the operation to complete its task, whether due to an error or an explicit cancellation.

Available in macOS 10.10 and newer in 64-bit applications.
Subclass of the CKDatabaseOperationMBS class.

6.29.2 Methods

6.29.3 Constructor(subscriptionsToSave() as CKSubscriptionMBS, subscriptionIDsToDelete() as String = nil)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to save and delete the specified subscriptions.
Notes: subscriptionsToSave: An array of CKSubscription objects representing the subscriptions to save or update. You may specify nil for this parameter.
subscriptionIDsToDelete: An array of strings representing the IDs of the subscriptions you want to delete, if any. You may specify nil for this parameter.

The subscriptions you intend to save or delete must all reside in the same container, which you must specify when configuring the operation object. Saving a subscription that is not in the current database creates it in the database. Trying to delete a subscription that does not exist in the current database returns an error for that subscription.

6.29.4 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.29.5 setSubscriptionIDsToDelete(SubscriptionIDsToDelete() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the IDs of the subscriptions you want to delete permanently from the database.

6.29.6 setSubscriptionsToSave(IDs() as CKSubscriptionMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the subscriptions to save to the database.

6.29.7 subscriptionIDsToDelete as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The IDs of the subscriptions you want to delete permanently from the database.
Notes: This property contains the array of strings that identify the subscriptions to delete. The initial contents of the array are set to the records you specified with the Constructor.

If you intend to change the set of subscriptions to be deleted, update the value of this property before executing the operation or submitting the operation object to a queue.

6.29.8 subscriptionsToSave as CKSubscriptionMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The subscriptions to save to the database.
Notes: This property contains the array of CKSubscriptionMBS objects that you want to save. The initial contents of the array are set to the subscriptions you specified with the Constructor. You can modify this array as needed before executing the operation. After saving the subscription objects, the server begins applying the criteria from those objects to the contents of the database, generating push notifications as appropriate.

If you intend to change the set of subscriptions to be saved, update the value of this property before executing the operation or submitting the operation object to a queue.

6.29.9 Events

6.29.10 modifySubscriptionsCompleted(savedSubscriptions() as CKSubscriptionMBS, deletedSubscriptionIDs() as String, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute after the status of all changes is known.
Notes: The event returns no value and takes the following parameters:

savedSubscriptions: The array of CKSubscription objects you tried to save.
deletedSubscriptionIDs: An array of NSString objects corresponding to the subscriptions you tried to delete.
operationError: An error object containing information about a problem, or nil if the results are saved successfully.

The operation object executes this block only once and represents your last chance to process the operation results. The block is executed on a background thread of your app. If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

This block reports an error of type CKErrorPartialFailure when it saves or delete only some of the subscriptions successfully. The userInfo dictionary of the error contains a CKPartialErrorsByItemIDKey key whose value is a dictionary. The keys of that dictionary are the IDs of the subscriptions that were not saved or deleted, and the corresponding values are error objects containing information about what happened.

6.30 class CKNotificationIDMBS

6.30.1 class CKNotificationIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKNotificationID object uniquely identifies a push notification sent from a container.
Notes: You do not create notification IDs directly. The server creates them when it creates the CKNotification objects that correspond to the push notifications sent to your app. You can compare two IDs using the isEqual method to determine if two notification objects are the same. This class defines no methods or properties.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.30.2 Methods

6.30.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.30.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.30.5 IsEqual(Other as CKNotificationIDMBS) as boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Checks if two objects are equal.

6.30.6 Properties

6.30.7 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.31 class CKNotificationInfoMBS

6.31.1 class CKNotificationInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKNotificationInfo object specifies the push notification data that the server sends to your app when a subscription finds a match.
Notes: When configuring a CKSubscriptionMBS object, create one of these objects and use it to specify the type of push notifications you want generated when the subscription,Äôs trigger condition is met. You can provide a message for an alert panel, information about the sounds to play, and information about whether the app should be badged. You can also ask the server to provide information about the record that triggered the notification.

see
https://developer.apple.com/reference/cloudkit/cknotificationinfo

6.31.2 Methods

6.31.3 alertLocalizationArgs as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The array of fields to use when building an alert message.
Notes: Use of this property is optional. This property contains an array of strings, each of which corresponds to a field of the record that triggered the push notification. Those names are used to retrieve the corresponding values from the record. The values themselves must be string, integer, double, or dates. Do not specify keys with other values. String values that are greater than 100 characters in length may be truncated when added to the push notification.

If you use %@ for your substitution variables, those variables are replaced by walking the array in order. If you use variables of the form %n$ @, where n is an integer, n represents the index (starting at 1) of the item in the array to use. Thus, the first item in the array replaces the variable %1$ @, the second item replaces the variable %2$ @, and so on. You can use indexed substitution variables to change the order of items in the resulting string, which might be necessary when you localize your app,Äôs messages.

6.31.4 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.31.5 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.31.6 copy as CKNotificationInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.31.7 desiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The names of fields to include in the push notification payload.
Notes: This property contains an array of strings, each of which corresponds to the name of a field in the record that triggered the notification. When a push notification is delivered, the keys and their corresponding values are included in the payload of the push notification. You can include a maximum of three keys in the array.

For the keys you specify, the allowable values are string, integer, double, CLLocationMBS, date, and CKReferenceMBS. You cannot specify keys whose values contain other data types. String values that are greater than 100 characters in length may be truncated when added to the push notification.

6.31.8 setAlertLocalizationArgs(args() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the array of fields to use when building an alert message.
Notes: Use of this property is optional. This property contains an array of strings, each of which corresponds to a field of the record that triggered the push notification. Those names are used to retrieve the corresponding values from the record. The values themselves must be string, integer, double, or dates. Do not specify keys with other values. String values that are greater than 100 characters in length may be truncated when added to the push notification.

If you use %@ for your substitution variables, those variables are replaced by walking the array in order. If you use variables of the form %n$ @, where n is an integer, n represents the index (starting at 1) of the item in the array to use. Thus, the first item in the array replaces the variable %1$ @, the second item replaces the variable %2$ @, and so on. You can use indexed substitution variables to change the order of items in

the resulting string, which might be necessary when you localize your app,Äôs messages.

6.31.9 setDesiredKeys(desiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the names of fields to include in the push notification payload.
Notes: This property contains an array of strings, each of which corresponds to the name of a field in the record that triggered the notification. When a push notification is delivered, the keys and their corresponding values are included in the payload of the push notification. You can include a maximum of three keys in the array.

For the keys you specify, the allowable values are string, integer, double, CLLocationMBS, date, and CKReferenceMBS. You cannot specify keys whose values contain other data types. String values that are greater than 100 characters in length may be truncated when added to the push notification.

6.31.10 Properties

6.31.11 alertActionLocalizationKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The key that identifies the text to use for the action button in the alert panel.
Notes: This property identifies the text to use for the button to open your app. Specifically, it contains the name of a key to look up in the app,Äôs Localizable.strings file, the value of which is used for the button title.

Specifying a value for this property is optional. When its value is nil, the alert panel triggered by the push notification contains only an OK button to dismiss the alert. When its value is not nil, the alert panel contains one button to dismiss the alert and a second button to open your app.
(Read and Write property)

6.31.12 alertBody as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The text to use for the alert message.
Notes: Use of this property is optional. If you set its value, the corresponding push notification causes the device to display an alert with the specified message. If you want to use a localized string for the alert message, specify a value for the alertLocalizationKey property instead.
(Read and Write property)

6.31.13 alertLaunchImage as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The filename of an image to use as a launch image.
Notes: Use of this property is optional. If you specify a value, the string is used to locate an image file in the app bundle. That image is displayed as a launch image when the user launches the app after receiving a push notification.
(Read and Write property)

6.31.14 alertLocalizationKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The key that identifies the localized string to use for the alert message.
Notes: Use of this property is optional. If you set its value, the corresponding push notification causes the device to display an alert on the user,Äôs device. The push notification obtains the text for the alert by looking up the specified key in your app,Äôs Localizable.strings file. If you specify a value for this property, the value in the alertBody property is ignored.

For information about localizing string resources, see Internationalization and Localization Guide.
(Read and Write property)

6.31.15 category as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Name of the action group corresponding to this notification.
Notes: Categories allow you to present custom actions to the user on your push notifications. For more information, read UIMutableUserNotificationCategory.
(Read and Write property)

6.31.16 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.31.17 shouldBadge as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean setting that controls whether a badge value should be incremented.
Notes: The default value of this property is false. Setting it to true causes the system to increment the current badge count for the app whenever the corresponding push notification is delivered.
(Read and Write property)

6.31.18 shouldSendContentAvailable as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the push notification should include the content-available flag.
Notes: When this property is true, the server includes the content-available flag in the push notification,Äôs payload. That flag causes the system to wake or launch an app that is not currently running. The app is then given background execution time to download any data related to the push notification, such as the set of records that changed. If the app is already running in the foreground, the inclusion of this flag has no additional effect and the notification is delivered to the app delegate for processing as usual.

The default value of this property is false.
(Read and Write property)

6.31.19 soundName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The name of the sound file to play when a notification arrives.
Notes: Use of this property is optional. If you specify a value, the string is used to locate a sound file in the app bundle. That sound file is played as an alert when a push notification arrives on the user,Äôs device. If the specified sound file does not exist, or if you specify the string default for this property, the system plays the default alert sound.
(Read and Write property)

6.32 class CKNotificationMBS

6.32.1 class CKNotificationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKNotificationMBS object represents a push notification that was sent to your app.
Notes: Notification objects wrap the data associated with a push notification. Use notification objects to wrap recently received push notification data or to fetch notification objects representing already delivered push notifications from a container. In both cases, the information in the notification object tells you what changed.

The CKNotificationMBS class itself is an abstract class. When you create a new notification object from a payload dictionary, the constructor instantiates a subclass of the appropriate type. Similarly, when you fetch notifications from a container, what you receive are instances of a concrete subclass. The base CKNotificationMBS class provides information about the push notification and how it was delivered. Specific subclasses contain specific data indicating what change actually occurred.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.32.2 Methods

6.32.3 alertLocalizationArgs as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The array of fields used to build the alert message
Notes: This property contains an array of NSString objects, each of which corresponds to a field of the record that triggered the push notification. Use the field values to replace any substitution variables in the alert strings specified by the notificationFromRemoteNotificationDictionary or Identifying the Notification Object properties. The field values must be string, integer, double, or dates. String values that are greater than 100 characters in length may be truncated when added to the push notification.

If you used %@ for your substitution variables, those variables are replaced by walking the array in order. If you use variables of the form %n$ @, where n is an integer, n represents the index (starting at 1) of the item in the array to use. Thus, the first item in the array replaces the variable %1$ @, the second item replaces the variable %2$ @, and so on. You can use indexed substitution variables to change the order of items in the resulting string, which might be necessary when you localize your app,Äôs messages.

6.32.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.32.5 notificationFromRemoteNotificationDictionary(notificationDictionary as Dictionary) as CKNotificationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates and returns a new notification object using the specified payload data.
Notes: notificationDictionary: The payload data for the push notification. This dictionary is the same one passed to your app delegate,Äôs didReceiveRemoteNotification method. This parameter must not be nil.

Returns a new notification object initialized with the payload data.

Use this method to initialize a notification object from a push notification received by your app.

6.32.6 Properties

6.32.7 alertActionLocalizationKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The key that identifies the text to use for the action button in the alert panel.
Notes: When this property is nil, the alert panel triggered by the push notification contains only an OK button to dismiss the alert. When this property is not nil, the alert panel contains one button to dismiss the alert and a second button to open your app. This property identifies the text to use for the button to open your app. Specifically, it contains the name of a key to look up in the app,Äôs Localizable.strings file, the value of which is used for the button title.
(Read only property)

6.32.8 alertBody as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The text of the alert message.
Notes: This property contains the nonlocalized version of the message string displayed by the alert.
(Read only property)

6.32.9 alertLaunchImage as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The filename of the launch image displayed when your app is launched from the push notification alert.
Notes: The string in this property is used to locate an image file in the app bundle.
(Read only property)

6.32.10 alertLocalizationKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The key that identifies the localized string to use for the alert message.
Notes: When delivered to your app, the push notification gets the text for the alert by looking up the specified key in your app,Äôs Localizable.strings file. When this property is set, the value in the notificationFromRemoteNotificationDictionary property is ignored.
(Read only property)

6.32.11 badge as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The current badge value.
Notes: The value of this property is the integer value displayed in the app,Äôs badge at the time the push notification was sent.
(Read only property)

6.32.12 category as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Name of the action group corresponding to this notification.
Notes: Categories allow you to present custom actions to the user on your push notifications. For more information, read UIMutableUserNotificationCategory.
(Read only property)

6.32.13 containerIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the container whose content triggered the notification.
Notes: (Read only property)

6.32.14 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.32.15 isPruned as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether some push notification content was removed prior to delivery.
Notes: see
https://developer.apple.com/reference/cloudkit/cknotification/1428088-ispruned
(Read only property)

6.32.16 notificationID as CKNotificationIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the notification.
Notes: (Read only property)

6.32.17 notificationType as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of event that generated the notification.
Notes: Different notification types correspond to different subclasses of CKNotificationMBS, so you can use the value in this property to determine how to handle the notification data.
(Read only property)

6.32.18 soundName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The name of the sound file to play when a notification arrives.
Notes: This property contains the string used to locate a sound file in your app,Äôs bundle. That sound file is played as an alert when a push notification arrives on the user,Äôs device. If the specified sound file does not exist, or if you specify the string default for this property, the system plays the default alert sound.
(Read only property)

6.32.19 subscriptionID as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The identifier of the subscription that caused this notification to fire.
Notes: (Read only property)

6.32.20 Constants

Notification Types

Constant	Value	Description
TypeDatabase	4	A notification generated when the contents of a database changed.
TypeQuery	1	A notification generated based on the conditions set forth in a subscription object.
TypeReadNotification	3	A notification that your app previously marked as read using a CKMarkNotificationsReadOperationMBS object.
TypeRecordZone	2	A notification generated when the contents of a record zone changed.

6.33 class CKOperationConfigurationMBS

6.33.1 class CKOperationConfigurationMBS

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: An object that describes how a CloudKit operation behaves.
Example:

```
Dim operationConfiguration As CKOperationConfigurationMBS = New CKOperationConfigurationMBS()
operationConfiguration.timeoutIntervalForRequest = 30
operationConfiguration.timeoutIntervalForResource = 30
operationConfiguration.QualityOfService = CKOperationConfigurationMBS.NSQualityOfServiceUserInitiated
```

Notes: Requires macOS 10.13, iOS 11.0 or newer.
Blog Entries

- News from the MBS Xojo Plugins Version 21.4
- MBS Xojo Plugins, version 21.4pr1

6.33.2 Methods

6.33.3 Constructor

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The constructor.

6.33.4 Properties

6.33.5 AllowsCellularAccess as Boolean

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: A Boolean value that indicates whether operations that use this configuration can send data over the cellular network.
Notes: (Read and Write property)

6.33.6 Container as CKContainerMBS

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The configuration,Äôs container.
Notes: If you don,Äôt provide a container, CloudKit uses the default container that CKContainerMBS provides.
(Read and Write property)

6.33.7 Handle as Integer

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.33.8 LongLived as Boolean

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: A Boolean value that indicates whether the operations that use this configuration are long-lived.
Notes: (Read and Write property)

6.33.9 QualityOfService as Integer

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The priority that the system uses when it allocates resources to the operations that use this configuration.
Notes: (Read and Write property)

6.33.10 timeoutIntervalForRequest as Double

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The maximum amount of time that a request can take.
Notes: In seconds.
(Read and Write property)

6.33.11 timeoutIntervalForResource as Double

Plugin Version: 21.4, Platforms: macOS, iOS, Targets: All.

Function: The maximum amount of time that a resource request can take.
Notes: In seconds.
(Read and Write property)

6.33.12 Constants

Quality of Service Constants

Constant	Value	Description
NSQualityOfServiceBackground	&h09	Background QoS is used for work that is not user initiated or visible. In gener a user is unaware that this work is even happening and it will run in the mo efficient manner while giving the most deference to higher QoS work. E example, pre-fetching content, search indexing, backups, and syncing of da with external systems.
NSQualityOfServiceDefault	-1	Default QoS indicates the absence of QoS information. Whenever possible Q information will be inferred from other sources. If such inference is not possib a QoS between UserInitiated and Utility will be used.
NSQualityOfServiceUserInitiated	&h19	UserInitiated QoS is used for performing work that has been explicitly r quested by the user and for which results must be immediately presented order to allow for further user interaction. For example, loading an email af a user has selected it in a message list.
NSQualityOfServiceUserInteractive	&h21	UserInteractive QoS is used for work directly involved in providing an intera tive UI such as processing events or drawing to the screen.
NSQualityOfServiceUtility	&h11	Utility QoS is used for performing work which the user is unlikely to be i mediately waiting for the results. This work may have been requested the user or initiated automatically, does not prevent the user from further i teraction, often operates at user-visible timescales and may have its progr indicated to the user by a non-modal progress indicator. This work will r in an energy-efficient manner, in deference to higher QoS work when resourc are constrained. For example, periodic content updates or bulk file operatio such as media import.

6.34 class CKOperationMBS

6.34.1 class CKOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The abstract base class for all operations that can be executed against a CloudKit database.
Notes: see
https://developer.apple.com/reference/cloudkit/ckoperation
Blog Entries

- MBS Xojo Plugins, version 21.4pr1
- MBS Xojo Plugins, version 17.3pr3

6.34.2 Methods

6.34.3 cancel

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Advises the operation object that it should stop executing its task.
Notes: This method does not force your operation code to stop. The code for your operation must invoke the isCancelled method periodically to determine whether the operation should be stopped. Once cancelled, an operation cannot be restarted.

If the operation is already finished executing, this method has no effect. Canceling an operation that is currently in an operation queue, but not yet executing, causes it to be removed from the queue (although not necessarily right away).

6.34.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.34.5 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.34.6 isCancelled as boolean

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the operation has been cancelled.
Notes: True if the operation was explicitly cancelled by an invocation of the operation's cancel method; otherwise, false. This method may return true even if the operation is currently executing.

Discussion
Canceling an operation does not actively stop the operation's code from executing. An operation object is responsible for calling this method periodically and stopping itself if the method returns true.

6.34.7 isExecuting as boolean

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the operation is currently executing.
Notes: True if the operation is executing; otherwise, false if the operation has not been started or is already finished.

6.34.8 isFinished as boolean

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the operation is done executing.
Notes: True if the operation is no longer executing; otherwise, false.

6.34.9 start

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Begins the execution of the operation.
Notes: The default implementation of this method configures the execution environment for a non-concurrent operation and invokes the operation's main method. As part of the default configuration, this method performs several checks to ensure that the non-concurrent operation can actually run and generates appropriate KVO notifications for each change in the operation's state. If the operation's operation has already been performed, was cancelled, or is not yet ready to run, this method throws an NSInvalidArgumentException exception. If the operation is to be performed on a separate thread, this method may return before the operation itself completes on the other thread.

6.34.10 Properties

6.34.11 allowsCellularAccess as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKOperationConfigurationMBS instead. **Function:** A Boolean value indicating whether the operation object may send data over the cell network.
Notes: When you send or receive many records, or when you send records with large assets, you might set this property to false to avoid consuming too much of the user,Äôs cellular data bandwidth. For operations involving only a few records, it is fine to leave this property set to true, which is the default.

When this property is set to false, the operation executes normally but fails if Wi-Fi is not available.
(Read and Write property)

6.34.12 configuration as CKOperationConfigurationMBS

Plugin Version: 21.4, Platform: macOS, Targets: All.

Function: The operation,Äôs configuration.
Notes: Requires macOS 10.13, iOS 11.0 or newer.
(Read and Write property)

6.34.13 container as CKContainerMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKOperationConfigurationMBS instead. **Function:** The container to use for the operation.
Notes: The container sets the context for where the operation should perform its work. The addOperation method of both the CKContainerMBS and CKDatabaseMBS classes implicitly sets the value of this property to their container.

If you execute the operation yourself, either directly or using a custom operation queue, it is recommended that you set the value of this property explicitly. If the value is nil when you execute an operation, the operation object implicitly executes against your app,Äôs default container.
(Read and Write property)

6.34.14 Handle as Integer

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: The handle to the internal used NSOperation reference.
Notes: (Read and Write property)

6.34.15 longLived as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKOperationConfigurationMBS instead. **Function:** A Boolean value indicating whether the operation is long-lived.
Notes: To create a long-lived operation, set to true. The default value is false. If the operation is running or is a long-lived operation fetched from a CKContainer object, changing this property value has no effect.

For more information on long-lived operations, read Long-Lived Operations.
(Read and Write property)

6.34.16 operationID as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A unique identifier for a long-lived operation.
Notes: Use this property to fetch a long-lived operation using the fetchLongLivedOperationWithID method in the CKContainer class. For more information on long-lived operations, read Long-Lived Operations.
(Read only property)

6.34.17 timeoutIntervalForRequest as Double

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKOperationConfigurationMBS instead. **Function:** The timeout interval to use when waiting for additional data.
Notes: This property determines the request timeout interval for the operation. The request timeout interval controls how long (in seconds) the operation should wait for additional data to arrive before giving up. The timer associated with this value is reset whenever new data arrives. When the request timer reaches the specified interval without receiving any new data, it triggers a timeout.

The default value is 60 seconds.
(Read and Write property)

6.34.18 timeoutIntervalForResource as Double

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use CKOperationConfigurationMBS instead. **Function:** The maximum amount of time that a resource request should be allowed to take.
Notes: This property determines the resource timeout interval for this operation. The resource timeout interval controls how long (in seconds) to wait for the entire operation to complete before giving up. The resource timer starts when the operation is initiated and counts until either the operation completes or this timeout interval is reached, whichever comes first.

The default value is 7 days.
(Read and Write property)

6.34.19 Events

6.34.20 Completed

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The operation completed.

6.34.21 LongLivedOperationWasPersisted

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute when the server starts storing callbacks for this long-lived operation.
Notes: If your app exits before this event is called, the long-lived operation identifier is not included in the results of the fetchAllLongLivedOperationIDs method.

For more information on long-lived operations, read Long-Lived Operations.

6.35 class CKQueryCursorMBS

6.35.1 class CKQueryCursorMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKQueryCursor object is an opaque data object that marks the stopping point for a query and the starting point for retrieving the remaining results.
Notes: You do not create instances of this class yourself. When fetching records using a CKQueryOperation object, if the number of results exceeds the results limit value set for the query, the server provides you with a query cursor object. Use that cursor object to initialize a new CKQueryOperationMBS and retrieve the next batch of results for the same query.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.35.2 Methods

6.35.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.35.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.35.5 copy as CKQueryCursorMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.35.6 Properties

6.35.7 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.36 class CKQueryMBS

6.36.1 class CKQueryMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKQuery object manages the criteria to apply when searching for records in a database.
Notes: You create a query object as the first step in the search process. The query object stores the search parameters, including the type of records to search, the match criteria (predicate) to apply, and the sort parameters to apply to the results. The second step is to use the query object to initialize a CKQueryOperation object, which you then execute to generate the results.

see
https://developer.apple.com/reference/cloudkit/ckquery

6.36.2 Methods

6.36.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.36.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.36.5 Constructor(RecordType as String, predicate as NSPredicateMBS) 345

6.36.5 Constructor(RecordType as String, predicate as NSPredicateMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a query object with the specified parameters.
Notes: recordType: The type of record to search. Specify the name of one of your app,Äôs supported record types. This method throws an exception if this parameter is nil or contains an empty string.
predicate: The search predicate to apply to the prospective records. Only records matching the predicate criteria are returned in the search results. For guidelines on how to construct predicates for your queries,

see Predicate Rules for Query Objects. This parameter must not be nil.

Returns an initialized query object.

Discussion
You cannot change the record type and predicate of a query object after you create it. If you want to search for a different set of records using a different set of search criteria, create a new query object. You can add sort descriptors to the query and change them later as needed.

You cannot query for user records and executing a query where the record type is set to CKRecordTypeUserRecordMBS results in an error. You must fetch user records directly using their ID.
See also:

- 6.36.4 Constructor 345

6.36.6 setSortDescriptors(sortDescriptors() as NSSortDescriptorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the sort descriptors to use when organizing the query results.

6.36.7 sortDescriptors as NSSortDescriptorMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The sort descriptors to use when organizing the query results.
Notes: This property contains an array of NSSortDescriptor objects. You can add sort descriptors to a query object and change them later as needed. Each sort descriptor contains a field name of the intended record type and information about whether to sort values in that field in ascending or descending order. The default value of this property is nil, which means that records are returned in an indeterminate order.

The order of the items in the array matches the order in which the sort descriptors are applied to the results. In other words, the first sort descriptor in the array is applied first, followed by the second sort descriptor if needed, and the third, and so on.

6.36.8 Properties

6.36.9 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.36.10 predicate as NSPredicateMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The search criteria to use when matching records.
Notes: A predicate contains one or more expressions that evaluate to true or false. Expressions are often value-based comparisons, but predicates support other types of operators, including string comparisons and aggregate operations. For guidelines on how to construct predicates for your queries, see Predicate Rules for Query Objects.

This property is set at initialization time and cannot be changed later.
(Read only property)

6.36.11 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type to search.
Notes: Searches return only records of the specified type. This property is set at initialization time and cannot be changed later.

The record type is an app-specific string that you use to distinguish among the records of your app. The records of a given type all represent different instances of the same information. For example, an employee record type might store the employee,Äôs name, phone number, and a reference to the employee,Äôs manager.
(Read only property)

6.37 class CKQueryNotificationMBS

6.37.1 class CKQueryNotificationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKQueryNotification object represents a push notification that was generated by a subscription object.
Notes: Subscription objects represent persistent queries on the server. When the server detects a change related to a subscription, it sends a push notification to the client that created the subscription and logs the push notification in the container. You use instances of this class to get information about the record involved in the push notification.

see
https://developer.apple.com/reference/cloudkit/ckquerynotification
Subclass of the CKNotificationMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.37.2 Methods

6.37.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.37.4 Properties

6.37.5 databaseScope as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The database scope.
Notes: (Read only property)

6.37.6 isPublicDatabase as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the record is in the public database.
Notes: The value of this property is true if the record is in the public database, or false if it is in the current

user,Äôs private database.
(Read only property)

6.37.7 queryNotificationReason as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The event that triggered the delivery of the push notification.
Notes: Subscription-initiated notifications are triggered by the creation, deletion, or updating of a single record. The record in question must match the predicate specified by the subscription object for an event to be triggered.
(Read only property)

6.37.8 recordFields as Dictionary

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A dictionary of the fields that changed.
Notes: For updated and newly created records, this property contains the desired keys requested by the subscription object. When you configure the CKNotificationInfoMBS object of a subscription object, you can specify the names of one or more fields in the desiredKeys property of that notification information object. When a push notification is triggered, the values for each of those keys is retrieved from the record and included in the push notification,Äôs payload, space permitting.

For query notification objects fetched from a container, all keys and values are present. For query notification objects generated from an incoming push notification, one or more keys and values may be missing. Push notification payloads are limited in size, and record fields are one of the first pieces of data to be excluded when that size limit is exceeded. For information about the order in which fields are removed, see the overview of this class.
(Read only property)

6.37.9 recordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The identifier of the record that was created, deleted, or updated.
Notes: Use this identifier to fetch the record from the container. Check the value of the isPublicDatabase property to determine which database to fetch it from.
(Read only property)

6.37.10 Constants

Reasons

Constant	Value	Description
ReasonRecordCreated	1	A record matching the subscription,Äôs predicate was created.
ReasonRecordDeleted	3	A record matching the subscription,Äôs predicate was deleted.
ReasonRecordUpdated	2	A record matching the subscription,Äôs predicate was updated.

6.38 class CKQueryOperationMBS

6.38.1 class CKQueryOperationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKQueryOperation object is a concrete operation that you can use to execute queries against a database.
Notes: A query operation takes the query parameters you provide and applies those parameters to the specified database and zone, delivering any matching records asynchronously to the blocks that you provide.

see
https://developer.apple.com/reference/cloudkit/ckqueryoperation
Subclass of the CKDatabaseOperationMBS class.
Blog Entries

- MBS Xojo Plugins, version 21.4pr1

6.38.2 Methods

6.38.3 CKQueryOperationMaximumResults as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A placeholder value representing the maximum number of results to retrieve.
Notes: The value of this constant does not correspond to the actual number of records. The actual maximum value is determined dynamically by the server based on various conditions.

6.38.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.
See also:

6.38.5 Constructor(query as CKQueryMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object configured to search for records in the specified zone.
Notes: query: The query to use during the search. This parameter must not be nil.

Returns an initialized operation object.

You can use the returned CKQueryOperationMBS object only once to perform a search, but the object you specify in the query parameter can be reused as needed. When executed, this query object performs a new search and returns the first batch of results. If there are more results available, you must create a separate query object using the provided cursor object.
See also:

- 6.38.4 Constructor 351
- 6.38.6 Constructor(queryCursor as CKQueryCursorMBS) 352

6.38.6 Constructor(queryCursor as CKQueryCursorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns an operation object that returns more results from a previous search.
Notes: cursor: The cursor object identifying the previous search. This value is passed to the completion block of the previous search.

Returns an initialized operation object ready to continue the search.

Use this method to initialize a query operation that retrieves the next batch of results from a previous search. When executing searches based on a cursor, do not cache cursor objects for a long time before using them. A cursor is not a snapshot of the previous search results; it stores a relative offset into the results list. An operation object created using a cursor performs a new search, sorts the new set of results, and uses the previous offset value to determine where the next batch of results starts.
See also:

- 6.38.4 Constructor 351
- 6.38.5 Constructor(query as CKQueryMBS) 351

6.38.7 desiredKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The fields to retrieve for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the search operation. The value is an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the specified keys are included in the returned record. If you specify nil, the operation retrieves all keys for the record. The default value of this

property is nil.

If you intend to specify a value, you must do so before executing the operation or submitting the operation object to a queue.

6.38.8 Destructor

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Destructor

6.38.9 setDesiredKeys(desiredKeys() as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets the fields to retrieve for the requested records.
Notes: Use this property to limit the amount of data retrieved for each record during the search operation. The value is an array of strings, each of which contains the name of a field from the target records. When you retrieve a given record, only fields whose names match one of the specified keys are included in the returned record. If you specify nil, the operation retrieves all keys for the record. The default value of this property is nil.

If you intend to specify a value, you must do so before executing the operation or submitting the operation object to a queue.

6.38.10 Properties

6.38.11 cursor as CKQueryCursorMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The data cursor to use for continuing the search.
Notes: You set the initial value of this property with the Constructor. When you use a cursor, the contents of the query property are ignored. The data cursor stored in this property is an opaque object that is provided to you by the server.

If you intend to specify or change the value in this property, do so before executing the operation or submitting the operation object to a queue.
(Read and Write property)

6.38.12 query as CKQueryMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The query to use for the search.
Notes: You set the initial value of this property with the Constructor method. When the value in the cursor property is nil, the query operation uses the CKQueryMBS object in this property to execute a new search and return the results to your completion handler. If the cursor value is not nil, the cursor is used instead.

If you intend to specify or change the value of this property, do so before executing the operation or submitting the operation object to a queue.
(Read and Write property)

6.38.13 resultsLimit as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The maximum number of records to return at one time.
Notes: For most queries, leave the value of this property set to the default value, which is represented by the CKQueryOperationMaximumResults constant. When using that value, the server chooses a limit that aims to provide an optimal number of results that returns as many records as possible while minimizing delays in receiving those records. However, if you know that you want to process a fixed number of results, change the value of this property accordingly.
(Read and Write property)

6.38.14 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the zone containing the records to search.
Notes: When set, the value of this property limits the scope of the search to the records in the specified zone. If no zone is specified, the search takes place on all zones.

When you initialize the operation object using the Constructor method, the value of this property is set to nil and any changes you make to the property are ignored. When the operation object is executed, the cursor object provides the zone information from the initial search that generated the cursor.
(Read and Write property)

6.38.15 Events

6.38.16 queryCompleted(cursor as CKQueryCursorMBS, operationError as NSErrorMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The block to execute with the search results.
Notes: The block returns no value and takes the following parameters:

cursor: A CKQueryCursorMBS object that indicates there are more results to fetch or nil if the results parameter contains all of the remaining search results. Use the provided object to initialize a new query operation object when you are ready to retrieve the next batch of results.
operationError: An error object containing information about a problem, or nil if the results are retrieved successfully.

This block is executed only once and represents your last chance to process the operation results. It is executed after all of the individual progress blocks but before the operation,Äôs completion block. The block is executed serially with respect to the other progress blocks of the operation. If you intend to use this block to process results, update the value of this property before executing the operation or submitting the operation object to a queue.

When the results of a query operation are known, the operation object uses this block to deliver the available set of records to your app. If the number of records exceeds the value in resultsLimit, the operation object provides an opaque data object in the cursor parameter of your block that you can use to retrieve the next batch of results. You must create a separate operation object using that cursor to get the next batch of results.

6.38.17 recordFetched(record as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event to execute for each record returned by the query.
Notes: record: A CKRecordMBS object matching the search criteria.

After identifying and sorting the records, the operation object executes this block once for each record in the sorted results. The block is executed serially with respect to all progress blocks of the operation object, so you can expect only one block at a time to be executing for this operation object.

If you intend to use this block to process results, set it before executing the operation or submitting the operation object to a queue.

Warning

Query indexes are updated asynchronously so they are not guaranteed to be current. If you query for records that you recently changed and not allow enough time for those changes to be processed, the query results may be incorrect. The results may not contain the correct records and the records may be out of order.

6.39 class CKQuerySubscriptionMBS

6.39.1 class CKQuerySubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Subscription that fires whenever a change matches the specified predicate.
Notes: Subclass of the CKSubscriptionMBS class.

6.39.2 Methods

6.39.3 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a query-based subscription that monitors records with the specified options.
Notes: recordType: The string that identifies the type of records to track. You are responsible for naming your app,Äôs record types. This parameter must not be nil or an empty string.
predicate: The matching criteria to apply to the records. This parameter must not be nil. For information about the operators that are supported in search predicates, see the discussion in CKQuery.
querySubscriptionOptions: A bitmask of the configuration options for the subscription. Specify at least one of the following values: CKSubscriptionOptionsFiresOnRecordCreation, CKSubscriptionOptionsFiresOnRecordUpdate, or CKSubscriptionOptionsFiresOnRecordDeletion.

Returns a subscription object initialized to track record-related changes.

The object returned by this method is configured as a query-based subscription searching records in the target dabatase. The subscription monitors the specified type of records in all of the user,Äôs record zones and generates push notifications when the search criteria are met.
See also:

- 6.39.4 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer) 357

6.39.4 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a query-based subscription that monitors records with the specified options.

Notes: recordType: The string that identifies the type of records to track. You are responsible for naming your app,Äôs record types. This parameter must not be nil or an empty string.
predicate: The matching criteria to apply to the records. This parameter must not be nil. For information about the operators that are supported in search predicates, see the discussion in CKQuery.
subscriptionID: The unique name of the subscription. This string must be unique in the specified database and must not be nil.
querySubscriptionOptions: A bitmask of the configuration options for the subscription. Specify at least one of the following values: CKSubscriptionOptionsFiresOnRecordCreation, CKSubscriptionOptionsFiresOnRecordUpdate, or CKSubscriptionOptionsFiresOnRecordDeletion.

Returns a subscription object initialized to track record-related changes.

The object returned by this method is configured as a query-based subscription searching records in the target dabatase. The subscription monitors the specified type of records in all of the user,Äôs record zones and generates push notifications when the search criteria are met.
See also:

- 6.39.3 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer) 357

6.39.5 copy as CKQuerySubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.39.6 Properties

6.39.7 predicate as NSPredicateMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The matching criteria to apply to records.
Notes: A query-based subscription uses its search predicate to identify potential matches for records. It combines the predicate information with the value in the querySubscriptionOptions property to determine the conditions under which to send a push notification to the app.

The search predicate defines the records that the subscription object monitors for changes. The value in this property is used only if the subscriptionType property is set to CKSubscriptionTypeQueryMBS; otherwise, it is ignored.
(Read only property)

6.39.8 querySubscriptionOptions as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Option flags describing the firing behavior of the subscription
Notes: Set the value of this property at initialization time. When configuring a query-based subscription, one of the following values must be specified:

CKQuerySubscriptionOptionsFiresOnRecordCreation
CKQuerySubscriptionOptionsFiresOnRecordUpdate
CKQuerySubscriptionsFiresOnRecordDeletion

If an option flag is not set, an NSInvalidArgumentException is thrown.
(Read only property)

6.39.9 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type being monitored in a query-based subscription. (read-only)
Notes: The value of this property applies only to query-based subscriptions and is set automatically by the Constructor. For all other types of subscription objects, the value of this property is ignored and set to nil.
(Read only property)

6.39.10 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Zone that this query subscription is scoped to.
Notes: The value of this property applies both to query-based subscriptions and zone-based subscriptions. Specifying a record zone limits the search scope to the records in that zone. In the case of a zone-based subscription, the search encompasses all records in the zone. For a query-based subscription, the search encompasses only records of a specific type in that zone.

For zone-based subscriptions, the value of this property is set automatically by the Constructor. For all other subscription types, the default value is nil. To apply a zone to a query-based subscriptions, you must assign a value explicitly.
(Read only property)

6.39.11 Constants

Option Flags

Constant	Value	Description
OptionsFiresOnce	8	Fire only once.
OptionsFiresOnRecordCreation	1	Fire at record creation.
OptionsFiresOnRecordDeletion	4	Fire on record deletion.
OptionsFiresOnRecordUpdate	2	Fire on record update.

6.40 class CKRecordIDMBS

6.40.1 class CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKRecordID object uniquely identifies a record in a database.
Notes: Record ID objects are normally created automatically when you create a new record, but you might also create IDs in several specific situations. For example, you must create record ID objects when you want to save a record in a zone other than the default zone. You can also create record ID objects when you want to retrieve records whose IDs you know from a database.

see
https://developer.apple.com/reference/cloudkit/ckrecordid

6.40.2 Methods

6.40.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.40.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.40.5 Constructor(recordName as string) 361
- 6.40.6 Constructor(recordName as string, zoneID as CKRecordZoneIDMBS) 362

6.40.5 Constructor(recordName as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a new record ID with the specified name in the default zone.
Notes: recordName: The name to use to identify the record. The string must contain only ASCII characters and must not exceed 255 characters. If you specify nil or an empty string for this parameter, this method

throws an exception.

Returns an initialized record ID object or nil if the object cannot be created.

Use this method when you are creating or searching for records in the default zone.
See also:

- 6.40.4 Constructor 361
- 6.40.6 Constructor(recordName as string, zoneID as CKRecordZoneIDMBS) 362

6.40.6 Constructor(recordName as string, zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a new record ID with the specified name and zone information.
Notes: recordName: The name to use to identify the record. The string must contain only ASCII characters and must not exceed 255 characters. If you specify nil or an empty string for this parameter, this method throws an exception.
zoneID: The ID of the zone in which to place the record. This parameter must not be nil.

Returns an initialized record ID object or nil if the object cannot be created.

Use this method when you create or search for records in a zone other than the default zone. The value in the zoneID parameter must represent a zone that already exists in the database. If the record zone does not exist yet, save the corresponding CKRecordZoneMBS object to the database before attempting to save any CKRecordMBS objects in that zone.
See also:

- 6.40.4 Constructor 361
- 6.40.5 Constructor(recordName as string) 361

6.40.7 copy as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.40.8 IsEqual(Other as CKRecordIDMBS) as boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Checks if two objects are equal.

6.40.9 Properties

6.40.10 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.40.11 recordName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique name of the record.
Notes: (Read only property)

6.40.12 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the zone containing the record.
Notes: (Read only property)

6.41 class CKRecordMBS

6.41.1 class CKRecordMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A dictionary of key-value pairs that you use to fetch and save the data of your app.
Notes: Records are the fundamental objects you use to manage data in CloudKit. You may define any number of record types for your app, with each record type corresponding to a different type of information you need. Within a given record type, you then define one or more fields, each of which has a name and a data value. Records can contain simple data types such as strings and numbers or more complex types such as geographic locations or pointers to other records.

An important step in using CloudKit is defining the record types your app supports. Each new record object contains no keys or values initially. During development, you can add new keys and values at any time. The first time you set a value for a key and save the record, the server associates that type with the key for all records of the same type. (The CKRecordMBS class does not enforce these type constraints or do any local validation of a record,Äôs contents; those constraints are enforced by the server when you save records.)

Note

The ability to add new keys is only possible during development. When you deploy to a production environment, the server returns an error when you try to specify an unknown record type or try to save a record containing unknown keys.
Although records act like dictionaries, there are still limitations to the types of values you can assign to keys. The following are the object types that the CKRecord class supports. Attempting to specify objects of any other type is a programmer error and will fail. Fields of all types are searchable unless otherwise noted.

see also
https://developer.apple.com/reference/cloudkit/ckrecord
Blog Entries

- MBS Xojo Plugins, version 21.5pr2
- MBS Xojo Plugins, version 21.3pr4
- News from the MBS Xojo Plugins Version 21.2
- MBS Xojo Plugins, version 21.2pr7

6.41.2 Methods

6.41.3 allKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of strings corresponding to all keys currently in the record.
Notes: Returns an array of strings. The returned array contains only the keys that have corresponding values in the record. If no keys are set for the record, this method returns an empty array.

This method may not return all possible keys in the record. Specifically, the method does not return keys whose values are nil.

6.41.4 allTokens as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of strings that you can use for full-text searches of the field,Äôs string-based values.
Notes: Returns an array of strings containing data from the records string-based fields.

When performing your own full-text searches, you can use this method to get a list of strings for your search. The method acts only on keys whose values are NSString objects. It breaks each value string apart at whitespace boundaries, creates new strings for each word, adds the new strings to an array, and returns the array. This tokenized version of the record,Äôs string values makes it easier to do string-based comparisons of individual words.

6.41.5 archive(byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Archives the content of this record via NSKeyedArchiver.
Example:

```
Dim ckRecord As New CKRecordMBS("MyType")

// archive
Dim e1 As NSErrorMBS
Dim encodedRecordValue As String = ckRecord.archive(e1)

// unarchive
Dim e2 As NSErrorMBS
Dim decodedRecord As CKRecordMBS = CKRecordMBS.Unarchive(encodedRecordValue, e2)

MsgBox "OK "+decodedRecord.recordType
```

Notes: In case of error returns nil and sets error.

6.41.6 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.41.7 changedKeys as String()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of strings representing the keys that have changed recently.
Notes: An array of strings, each of which represents a key whose value has changed since the record was downloaded or saved. If no keys have changed, this method returns an empty array.

6.41.8 CKRecordTypeUserRecord as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type for an user record.

6.41.9 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

6.41.10 Constructor(Coder as NSCoderMBS)

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Initializes record by reading from coder.
Notes: e.g. use with NSKeyedUnarchiverMBS class.
See also:

6.41.11 Constructor(RecordType as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a new record of the specified type.
Example:

```
// Create a new record of type "employee".
dim myRecord as new CKRecordMBS("employee")
```

Notes: recordType: A string reflecting the type of record that you want to create. This string becomes the type of record thereafter and cannot be changed. You define the record types that your app supports and use them to distinguish between records with different types of data. This parameter must not be nil or contain an empty string.

Record type names must consist of one or more alphanumeric characters and must start with a letter. Type names may include underscore characters as long as they do not start with that character. Spaces are not allowed in the names.

Returns an initialized record object or nil if the record cannot be created.

Use this method to initialize a new record object in the default zone of the database. The newly created record contains no data in any of its fields and is assigned a unique ID.

New records exist only in memory until you explicitly save them to iCloud. In addition, new records are sparse by default and have no values assigned to the fields you defined. (In fact, until you set the value of a key explicitly, getting the value of a key in a new record returns nil.) Even though a record has an associated type, that type information is ignored until you save the record.

Save the record using a CKModifyRecordsOperationMBS object or using the saveRecord method of CK-DatabaseMBS to transfer the record,Äôs contents to the server.
See also:

6.41.12 Constructor(RecordType as String, recordID as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a record using an ID that you provide.
Notes: recordType: A string reflecting the type of record that you want to create. Define the record types that your app supports, and use them to distinguish between records with different types of data. This parameter must not be nil or contain an empty string. Record type names consist of one or more alphanumeric characters and start with a letter. Type names may include underscore characters if they do not start with that character. Spaces are not allowed in record type names.
recordID: The ID to assign to the record itself. When creating the ID, you can specify the zone in which to place the record. The ID cannot currently be in use by any other record and must not be nil.

Returns an initialized record object or nil if the record cannot be created.

Discussion
Use this method to initialize a new record object with the specified ID. The newly created record contains no data.

Upon creation, record objects exist only in memory on the local device. Save the record using a CKModifyRecordsOperationMBS object or using the saveRecord method of CKDatabaseMBS to transfer the record,Äôs contents to the server.
See also:

6.41.13 Constructor(RecordType as String, zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a record in the specified zone.
Notes: recordType: A string reflecting the type of record that you want to create. Define the record types that your app supports, and use them to distinguish between records with different types of data. This parameter must not be nil or contain an empty string. Record type names consist of one or more alphanumeric characters and start with a letter. Type names may include underscore characters if they do not start with that character. Spaces are not allowed in record type names.

zoneID: The ID of the record zone in which to place the record.

Returns an initialized record object, or nil if the record cannot be created.

Use this method to initialize a new record object in the specified record zone.

Upon creation, the new record contains no data and exists only in memory on the local device. Save the record using a CKModifyRecordsOperationMBS object or using the saveRecord:completion: method of CKDatabase to transfer the record,Äôs contents to the server.
See also:

6.41.14 copy as CKRecordMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.41.15 encodeSystemFieldsWithCoder(Coder as NSCoderMBS)

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Encodes the record,Äôs system fields using the specified archiver.
Notes: Use this method to encode the record,Äôs metadata that CloudKit provides. Every record has keys that the system defines that correspond to record metadata, such as the record ID, record type, creation date, and so on. This method encodes those keys in the specified archiver. This method doesn,Äôt include any keys you add to the record. It also doesn,Äôt encode the keys that the changedKeys() method returns. You might use this method when you want to store only the system metadata because you store the actual record data elsewhere.

use with NSKeyedArchiverMBS class.

6.41.16 encodeWithCoder(Coder as NSCoderMBS)

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Encodes this record with the given coder.
Notes: Best pass instance of NSKeyedArchiverMBS to encode.

6.41.17 setParentReferenceFromRecord(parentRecord as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates and sets a reference object for a parent from its record.
Notes: parentRecord: A record that you want to set as the parent to this record.

This method creates and sets a CKReferenceMBS object for the CKRecordMBS passed in. The resulting CKReferenceMBS will have its action set to none.

6.41.18 setParentReferenceFromRecordID(parentRecordID as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Sets parent via Record ID.

6.41.19 Unarchive(Data as MemoryBlock, byref error as NSErrorMBS) as CKRecordMBS

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Archives the data of to a new CKRecord via NSKeyedUnarchiver.
Example:

```
Dim ckRecord As New CKRecordMBS("MyType")

// archive
Dim e1 As NSErrorMBS
Dim encodedRecordValue As String = ckRecord.archive(e1)

// unarchive
Dim e2 As NSErrorMBS
Dim decodedRecord As CKRecordMBS = CKRecordMBS.Unarchive(encodedRecordValue, e2)

MsgBox "OK "+decodedRecord.recordType
```

Notes: In case of error returns nil and sets error.

6.41.20 Properties

6.41.21 creationDate as Date

Plugin Version: 16.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The time when the record was first saved to the server.
Notes: The creation date reflects the time at which a record with the current record,Äôs ID was created on the server. For new instances of this class, the value of this property is initially set to nil. When you save the record to the server, the value is updated with the appropriate creation date for the record.
(Read only property)

6.41.22 creationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The time when the record was first saved to the server.
Notes: The creation date reflects the time at which a record with the current record,Äôs ID was created on the server. For new instances of this class, the value of this property is initially set to nil. When you save the record to the server, the value is updated with the appropriate creation date for the record.
(Read only property)

6.41.23 creatorUserRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the user who created the record.
Notes: Use the value in this property to retrieve the user record of the user who created this record. Every user of the app has a unique user record that is empty by default. Apps can add data to the user record on behalf of the user but should not store sensitive data in it.
(Read only property)

6.41.24 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.41.25 lastModifiedUserRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the user who last modified the record.
Notes: Use the value in this property to retrieve the user record of the user who last modified this record. Every user of the app has a unique user record that is empty by default. Apps can add data to the user record on behalf of the user but should not store sensitive data in it.
(Read only property)

6.41.26 modificationDate as Date

Plugin Version: 16.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The time when the record was last saved to the server.
Notes: The modification date reflects the time at which a record with the current record,Äôs ID was last saved to the server. For new instances of this class, the value of this property is initially set to nil. When you save the record to the server, the value is updated with the appropriate modification date for the record.
(Read only property)

6.41.27 modificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The time when the record was last saved to the server.
Notes: The modification date reflects the time at which a record with the current record,Äôs ID was last saved to the server. For new instances of this class, the value of this property is initially set to nil. When you save the record to the server, the value is updated with the appropriate modification date for the record.
(Read only property)

6.41.28 parent as CKReferenceMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to the parent record to this record.
Notes: A parent reference is used to teach CloudKit about the hierarchy of your records. This hierarchy of records will be shared if the share reference is set on a record.

A parent record reference must have a none set.

The target of a parent reference must exist at save time —either already on the server, or part of the same CKModifyRecordsOperationMBS batch.

You are encouraged to set up the parent relationships as part of normal record saves, even if you are not planning on sharing records at this time. This allows you to share and unshare a hiearchy of records at a later date by only modifying the ,Äútop level,Äù record, setting or clear its share reference.
(Read and Write property)

6.41.29 recordChangeTag as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A string containing the server change token for the record.
Notes: When you fetch a record from the server, you get the current version of that record as it exists on the server. However, at any time after you fetch a record, other users might save a newer version of the record to the server. Every time a record is saved, the server updates the record,Äôs change token to a new value. When you save your instance of the record to the server, the server compares the token in your record with the token on the server. If the two tokens match, the server knows that you modified the latest version of the record and that your changes can be applied right away. If the two tokens do not match, the server applies the save policy your app specified to determine how to proceed.

In your own code, you can use change tokens to distinguish between two different versions of the same record.
(Read only property)

6.41.30 recordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique ID of the record.
Notes: The ID of a new record is always set at initialization time. If you use the Constructor method to initialize the record, the ID is derived from the CKRecordIDMBS object you provide. In all other cases, the record generates a UUID and bases its ID on that value. The ID of a record never changes over the lifetime of that record.

When you save a new record object to the server, the server validates the uniqueness of the record but reports an error only if the save policy calls for it. Specifically, it reports an error when the save policy is set to ifServerRecordUnchanged, which is the default. For other save policies, the server overwrites the contents of the existing record accordingly.
(Read only property)

6.41.31 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The app-defined string that identifies the type of the record.
Notes: Use this string to differentiate between different record types in your app. The string is primarily for your benefit, so choose type names that reflect the data in the corresponding records.
(Read only property)

6.41.32 share as CKReferenceMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to the share associated with the sharing of this record.
Notes: The share property (CKReferenceMBS) on a record will be removed when the corresponding CKShare object is deleted from the server. Send this record in the same batch as the share that is being deleted, and this record,Äôs share property will be updated.

Sharing is only supported in zones with the CKRecordZoneCapabilitySharing capability. The default zone does not support sharing.

If any records have a parent reference to this record, they are implicitly shared alongside this record.

Note

Records in a hierarchy must only exist within one share. If a child record in a hierarchy already has a share reference set, you will get a CKErrorAlreadyShared error if you try to share any of that record,Äôs parents.
(Read only property)

6.41.33 dataForKey(key as string) as MemoryBlock

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The value for the given key stored in the record as data.
Notes: If assign nil as value, CloudKit removes any object that the record associates with the key.
(Read and Write computed property)

6.41.34 objectForKey(key as string) as Variant

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The value for the given key stored in the record.
Notes: key: The string that identifies a field in the record. Key names consist of one or more alphanumeric characters and start with a letter. You may also include underscore characters if you do not use an under-

score as the first character in the name. Spaces are not allowed in key names.

Returns the object associated with the specified key or nil if no such key exists in the record.

New records do not contain any keys or values initially. Values are always one of the object types listed in Supported Data Types.

You access the fields of a CKRecord object the same way you access key-value pairs in an NSMutableDictionary. The CKRecord class defines the objectForKey: and setObject:forKey: methods for getting and setting values. It also supports dictionary index notation.

If assign nil as value, CloudKit removes any object that the record associates with the key.
(Read and Write computed property)

6.41.35 stringForKey(key as string) as string

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The value for the given key stored in the record as string.
Notes: If field contains a number, we convert it to string for you.

If assign "" as value, we put in an empty string. To clear the key, pass nil via objectForKey assignment.
(Read and Write computed property)

6.42 class CKRecordZoneIDMBS

6.42.1 class CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKRecordZoneID object uniquely identifies a record zone in a database.
Notes: Zones are a way to group related records together. You create zone ID objects when you want to fetch an existing zone object or create a new zone with a specific name.

Overview
A record zone ID distinguishes one zone from another by a name string and the ID of the user that created the zone. Both strings must be ASCII strings that do not exceed 255 characters. For automatically created record zones, the ID name string is based on a UUID and is therefore guaranteed to be unique. When creating your own record zone ID objects, you are free to use names that have more meaning to your app or to the user, providing each zone name is unique within the specified database. The owner name must be either the current user name (obtained from the fetchUserRecordID method) or the name of another user.

When creating new record zones, make the name string in the record zone ID unique in the target database. Public databases do not support custom zones, and zones in a private database can only be created by the user that owns the database.

This class is not intended to be subclassed.

Interacting with Record Zone IDs
After you create a CKRecordZoneIDMBS object, interactions with that object typically include:

- Creating a CKRecordIDMBS object so that you can fetch or create records in that zone.
- Retrieving an existing CKRecordZoneMBS object from the database.

You do not need to create a CKRecordZoneIDMBS object in order to create a CKRecordZoneMBS object. The CKRecordZoneMBS class has initialization methods that create a record zone ID using the name string you provide.

Creating Record Zone IDs to Use with Records
To create a new record in a custom zone, first create a CKRecordZoneIDMBS object that specifies the zone name. Use the record zone ID to create a CKRecordID and then use the record ID to create the record itself.

Fetching a Record Zone Object from the Database
To fetch a CKRecordZoneMBS object from a database, use a CKFetchRecordZonesOperationMBS object or the fetchRecordZoneWithID method of the CKDatabaseMBS class. Both techniques take a CKRecord-ZoneIDMBS object that you provide and retrieve the corresponding record zone object asynchronously. If

you use the operation object, you can retrieve multiple record zones at the same time.

6.42.2 Methods

6.42.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.42.4 Constructor(zoneName as string, ownerName as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a record zone ID with the specified name and owner.
Notes: zoneName: The name that identifies the record zone. The string must contain only ASCII characters and must not exceed 255 characters. To specify the default zone of the current database, pass the CKRecordZoneDefaultName constant for this parameter. This parameter must not be nil or the empty string.
ownerName: The user who created the record zone. To specify the current user, use the CKOwnerDefaultName constant. If you specify nil or an empty string for this parameter, this method throws an exception.

Returns an initialized record zone ID object or nil if the object cannot be created.

Use this method to create a record zone ID for use in creating or fetching a record zone.

6.42.5 copy as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.42.6 IsEqual(Other as CKRecordZoneIDMBS) as boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Checks if two objects are equal.

6.42.7 Properties

6.42.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.42.9 ownerName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the user who owns the record zone.
Notes: (Read only property)

6.42.10 zoneName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique name of the record zone.
Notes: (Read only property)

6.43 class CKRecordZoneMBS

6.43.1 class CKRecordZoneMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A definition of a custom area for organizing related records in a database.
Notes: Zones are an important part of how you organize your data. The public and private databases have a single default zone. In the private database you can use CKRecordZoneMBS objects to create additional custom zones as needed. Use custom zones to arrange and encapsulate groups of related records in the private database. Custom zones support other capabilities too, such as the ability to write multiple records as a single atomic transaction.

Treat each custom zone as a single unit of data that is separate from every other zone in the database. Inside the zone, you add records as you would anywhere else. You can also create links between the records inside a zone by using the CKReferenceMBS class. However, the CKReferenceMBS class does not support cross-zone linking, so each reference object must point to a record in the same zone as the current record.

Use the CKRecordZoneMBS class as-is and do not subclass.

Creating a Custom Record Zone
For the most part, you use instances of this class to create and manage custom zones. Although you can use this class to retrieve a database,Äôs default zone, most operations act on records in the default zone by default, so you rarely need to specify it explicitly.

To create a custom zone, use CKRecordZoneMBS to create the zone object, and then save that zone to the user,Äôs private database using a CKModifyRecordZonesOperationMBS object. You cannot save any records in the zone until you save it to the database. When creating records, explicitly specify the zone ID if you want the records to reside in a specific zone; otherwise, they will be saved to the default zone. You cannot create custom zones in a public database.

After creating a CKRecordZoneMBS object and saving it to the database, you do not interact with the object much. Instead, most interactions occur with its associated CKRecordZoneIDMBS object, which you use to refer to the zone when creating records.

6.43.2 Methods

6.43.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.43.4 CKRecordZoneDefaultName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The name of the default zone.
Notes: Use this constant when you need to refer to the default zone by name, perhaps when creating a zone ID. The default zone has no special capabilities.

6.43.5 Constructor(zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a record zone object using the specified ID.
Notes: zoneID: The ID for the new zone. This parameter must not be nil.

Returns the new custom zone, or nil if the zone cannot be created.

Use this method when you want to create a new record zone based on the information in a zone ID. After creating the zone, save it to the server using a CKModifyRecordZonesOperationMBS object or the save method of the CKDatabaseMBS class.

Do not use this method to create a CKRecordZoneMBS object corresponding to a zone that already exists in the database. If the zone exists, fetch it using a CKFetchRecordZonesOperationMBS object or the fetch method of the CKDatabaseMBS class.
See also:

- 6.43.6 Constructor(zoneName as string) 380

6.43.6 Constructor(zoneName as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a record zone object with the specified name.
Notes: zoneName: The name of the new zone. Zone names inside a user,Äôs private database are unique, consist of ASCII characters, are no longer than 255 characters, and do not start with an underscore (_) character. One way to ensure the uniqueness of zone names is to create a string based on a UUID, but you can also use other techniques.

If this parameter is nil or is an empty string, the method throws an exception.

Returns the new custom zone, or nil if the zone cannot be created.

Use this method to create a new record zone. The new zone has the name you provide and the zone,Äôs owner is set to the current user. After creating the zone, save it to the server using a CKModifyRecordZonesOperationMBS object or the saveRecordZone method of the CKDatabase class. You must save the zone to the server before attempting to save any records to that zone.

Do not use this method to create a CKRecordZone object corresponding to a zone that already exists in the database. If the zone exists, fetch it using a CKFetchRecordZonesOperationMBS object or the fetchRecordZoneWithID method of the CKDatabase class.
See also:

- 6.43.5 Constructor(zoneID as CKRecordZoneIDMBS) 380

6.43.7 copy as CKRecordZoneMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.43.8 defaultRecordZone as CKRecordZoneMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns the default zone for records.
Notes: The default record zone for a database.

Always use this method to retrieve the default zone for a database. The returned object can be used to specify the default zone for either the public or private database of a container. You do not need to save the returned zone object before using it. The owner of the zone is set to CKOwnerDefaultName, which corresponds to the current user.

The default zone of a database is a convenient place to store and access records. Whenever you do not explicitly assign a zone to a record, CloudKit puts the record in the default zone.

The disadvantage of using the default zone for storing records is that it does not have any special capabilities. You cannot save a group of records to iCloud atomically in the default zone. Similarly, you cannot use a CKFetchRecordChangesOperationMBS object on records in the default zone.

6.43.9 Properties

6.43.10 capabilities as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The capabilities supported by the zone.
Notes: The server determines the capabilities of the zone and sets the value of this property when you save the record zone. Always check this property before performing tasks that require a specific capability.

Typically, default zones do not support any special capabilities. Custom zones in a private database normally support all options.

For more information about what you can do with specific capabilities, see Capabilities* constants.
(Read only property)

6.43.11 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.43.12 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique ID of the zone.
Notes: The zone ID contains the name of the zone and the name of the user who owns the zone. Use this property to access both of those values.
(Read only property)

6.43.13 Constants

Capabilities

Constant	Value	Description
CapabilityAtomic	2	A capability that allows changes to multiple records to be made atomically. When you use a CKModifyRecordsOperationMBS object to save records, if the server is unable to save the changes for one record, it does not save the changes for any of the records. When combined with the CKRecordSaveIfServerRecordUnchanged policy of the operation object, this behavior prevents your app from overwriting changes to a group of records if one of the records was modified elsewhere.
CapabilityFetchChanges	1	A capability that allows you to fetch only the changed values in records. This capability makes the creation of offline caches more efficient. Instead of fetching the entire record every time, use a CKFetchRecordChangesOperationMBS object to fetch only the values that changed, and use the returned data to update your caches. Doing so minimizes the amount of data you receive from the server. You can still fetch the entire record using a CKFetchRecordsOperationMBS object if you want.
CapabilitySharing	4	Sharing

6.44 class CKRecordZoneNotificationMBS

6.44.1 class CKRecordZoneNotificationMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A notification that is caused by changes to the contents of a record zone.
Notes: Zone-related changes occur when existing records in the zone are modified, when new records are added, and when existing records are deleted. Use instances of this class to determine which zone triggered the notification.

You do not create instances of this class directly. When your app receives a push notification, call the notificationFromRemoteNotificationDictionary method of CKNotificationMBS to generate a notification object of the appropriate type. You can also fetch previously delivered notifications from a container using a CKFetchNotificationChangesOperationMBS object. If the notification was triggered due to a record zone change, the operation object delivers an instance of CKRecordZoneNotificationMBS with its notificationType property set to CKNotificationTypeRecordZoneMBS. Use the record zone notification object to get the record zone information and other push-related data.

If a push notification,Äôs payload is too large, pieces of data may be dropped until the payload meets the allowed size limit. The data that alerts the user is the most important and is dropped last. Data values in this class are among the first to be dropped, with values being dropped in the following order:

1. containerIdentifier—defined in the CKNotification class.
2. recordZoneID
3. Other properties of the CKNotification class.

Subclass of the CKNotificationMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.44.2 Methods

6.44.3 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.44.4 Properties

6.44.5 databaseScope as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of database (public, private, or shared) associated with the zone.
Notes: For more details on the values returned, see CKDatabaseMBS.Scope* constants.
(Read only property)

6.44.6 recordZoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the zone that changed.
Notes: Use the value of this property to fetch the corresponding zone from the database.
(Read only property)

6.45 class CKRecordZoneSubscriptionMBS

6.45.1 class CKRecordZoneSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A subscription that causes a push notification to fire whenever any change happens in the specified record zone.
Notes: Subclass of the CKSubscriptionMBS class.

6.45.2 Methods

6.45.3 Constructor(zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a subscription object that monitors all records in the specified record zone.
Notes: zoneID: The ID of the record zone containing the records you want to monitor. This parameter must not be nil.

Returns a subscription object initialized to track changes to the contents of the specified record zone.

The object returned by this method is configured as a zone-based subscription, which generates a push notification when any changes are made to the records in the specified zone.
See also:

- 6.45.4 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string) 386

6.45.4 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a subscription object that monitors the specified zone and has a custom name that you provide.
Notes: zoneID: The ID of the record zone containing the records you want to monitor. This parameter must not be nil.
subscriptionID: The unique name of the subscription object. This string must be unique for each subscription object in the container. This parameter must not be nil.

Returns a subscription object initialized to track changes to the contents of a record zone.

The CKRecordZoneSubscriptionMBS returned by this method is configured as a zone-based subscription,

which generates a push notification when any changes are made to the records in the specified zone.
See also:

- 6.45.3 Constructor(zoneID as CKRecordZoneIDMBS) 386

6.45.5 copy as CKRecordZoneSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.45.6 Properties

6.45.7 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type being monitored in a query-based subscription. (read-only)
Notes: The value of this property applies only to query-based subscriptions and is set automatically by the CKRecordZoneSubscriptionMBS and CKRecordZoneSubscriptionMBS methods. For all other types of subscription objects, the value of this property is ignored and set to nil.
(Read and Write property)

6.45.8 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Zone that the specified query subscription is scoped to.
Notes: The value of this property applies both to query-based subscriptions and zone-based subscriptions. Specifying a record zone limits the search scope to the records in that zone. In the case of a zone-based subscription, the search encompasses all records in the zone. For a query-based subscription, the search encompasses only records of recordType in that zone.

For zone-based subscriptions, the value of this property is set automatically by the CKRecordZoneSubscriptionMBS or CKRecordZoneSubscriptionMBS methods. For all other subscription types, the default value is nil. To apply a zone to a query-based subscription, you must assign a value explicitly.
(Read only property)

6.46 class CKReferenceMBS

6.46.1 class CKReferenceMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKReference object creates a many-to-one relationship between records in your database.
Notes: Each reference object stores information about the one record that is the target of the reference. You then save the reference object in the fields of one or more records to create a link from those records to the target. Both records must be located in the same zone of the same database.

see
https://developer.apple.com/reference/cloudkit/ckreference

6.46.2 Methods

6.46.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.46.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.46.5 Constructor(record as CKRecordMBS, action as Integer = 0) 388
- 6.46.6 Constructor(recordID as CKRecordIDMBS, action as Integer = 0) 389

6.46.5 Constructor(record as CKRecordMBS, action as Integer = 0)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a reference object that points to the specified record object.
Notes: record: The target record of the reference.
action: The ownership options to put in place for the records. If you specify the deleteSelf option, the object referred to by the recordID parameter becomes the owner of any objects that use this reference object. For

a list of possible values, see CKReferenceAction.

Returns an initialized reference object that points to the specified record, or nil if the reference cannot be initialized.

Use this method to initialize a reference to a local record object. The local record can be one that you just created or one that you fetched previously from the server.

When you create a reference object for use in a search predicate, the predicate ignores the value in the action parameter. Search predicates use only the ID of the record during their comparison.
See also:

- 6.46.4 Constructor 388
- 6.46.6 Constructor(recordID as CKRecordIDMBS, action as Integer = 0) 389

6.46.6 Constructor(recordID as CKRecordIDMBS, action as Integer = 0)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a reference object that points to the record with the specified ID.
Notes: recordID: The ID of the target record. This method throws an exception if you specify nil for this parameter.
action: The ownership option to put in place between the target record and any records that incorporate this reference object. If you specify the deleteSelf option, the record referred to by the recordID parameter owns (or acts as the parent) of any objects that use this reference object. For a list of possible values, see CKReferenceActionMBS.

Returns an initialized reference object that points to the specified record, or nil if the reference cannot be initialized.

Use this method when you have only the ID of the record that is to become the target of a link. You might use this method if you saved only the ID of the record to a local data cache.

When you create a reference object for use in a search predicate, the predicate ignores the value in the action parameter. Search predicates use only the ID of the record during their comparison.
See also:

- 6.46.4 Constructor 388
- 6.46.5 Constructor(record as CKRecordMBS, action as Integer = 0) 388

6.46.7 copy as CKReferenceMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.46.8 Properties

6.46.9 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.46.10 recordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the referenced record.
Notes: Use the ID in this property to fetch the record on the other end of the link.
(Read only property)

6.46.11 referenceAction as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ownership behavior for the records.
Notes: The value in this property determines what action, if any, to take when the target of the reference object—that is, the object pointed to in the recordID property—is deleted. When this property is set to deleteSelf, deleting the target object deletes any records that contain that reference in one of their fields. When this property is set to none, deleting the target object does not delete any additional objects.
(Read only property)

6.46.12 Constants

Actions

Constant	Value	Description
ActionDeleteSelf	1	The delete action for referenced records. Deleting a record also deletes any records containing CKReferenceMBS objects that point to that record. The deletion of the additional records may trigger a cascade deletion of more records. The deletions are asynchronous in the default zone and immediate in a custom zone.
ActionNone	0	No action when a referenced record is deleted. Deleting a parent record does not delete the children that refer to that parent. The CKReference object still contains the ID of the deleted record and is not updated automatically.

6.47 class CKServerChangeTokenMBS

6.47.1 class CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A CKServerChangeTokenMBS object is an opaque data object that identifies a specific version of a record.
Notes: You do not create instances of this class yourself. When fetching records using a CKFetchRecordChangesOperation object, the server provides one of these objects along with the record changes. The next time you fetch the records, pass the previous token to the server. Passing the previous token tells the server what portions of the records to fetch and return to your app.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 21.2
- MBS Xojo Plugins, version 21.2pr5

6.47.2 Methods

6.47.3 Archive(byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Archives the token.
Notes: You may do Base64 on top of the memoryblock before storing in e.g. a XML or JSON block.
Uses NSKeyedArchiver internally.

6.47.4 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.47.5 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.47.6 copy as CKServerChangeTokenMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.47.7 IsEqual(Other as CKServerChangeTokenMBS) as boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Checks if two objects are equal.

6.47.8 Unarchive(Data as MemoryBlock, byref error as NSErrorMBS) as CKServerChangeTokenMBS

Plugin Version: 21.2, Platform: macOS, Targets: All.

Function: Unarchives the token.
Notes: If you used Base64 previously, don't forget to do Base64 Decode before passing in data.
Uses NSKeyedUnarchiver internally.

6.47.9 Properties

6.47.10 description as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The description text.
Notes: (Read only property)

6.47.11 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.48 class CKShareMBS

6.48.1 class CKShareMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to a shared record.
Notes: Available in macOS 10.12 and newer.
Subclass of the CKRecordMBS class.
Blog Entries

- MBS Xojo Plugins, version 21.5pr2
- MBS Xojo Plugins, version 21.3pr4

6.48.2 Methods

6.48.3 addParticipant(participant as CKShareParticipantMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Adds a participant to this share.
Notes: participant: The participant to be added to this share.

If a participant with a matching userIdentity already exists in this share, that existing participant,Äôs properties are updated; no new participant is added.

To modify the list of participants, a share must have publicPermission set to CKShareParticipantPermissionNone. You cannot mix and match private users and public users in the same share. Only certain participant types may be added via this API; see CKShareParticipantMBS.

6.48.4 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.48.5 CKRecordTypeShare as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of record for a share.

6.48.6 CKShareThumbnailImageDataKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the predefined keys in the CKRecordTypeShare schema.
Notes: They're used by the out of process UI flow to send a share, and as part of the share acceptance flow. These are optional.

Value is a data blob suitable to pass into NSImageMBS.imageWithData.

6.48.7 CKShareTitleKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the predefined keys in the CKRecordTypeShare schema.
Notes: They're used by the out of process UI flow to send a share, and as part of the share acceptance flow. These are optional.

Value is a string. Example for a recipe sharing app: "Pot Roast"

6.48.8 CKShareTypeKey as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: One of the predefined keys in the CKRecordTypeShare schema.
Notes: They're used by the out of process UI flow to send a share, and as part of the share acceptance flow. These are optional.

Value is a string representing a UTI. Example for a recipe sharing app: "com.mycompany.recipe"

6.48.9 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.48.10 Constructor(RecordType as String) 396

6.48.10 Constructor(RecordType as String)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

6.48.11 Constructor(RecordType as String, recordID as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

6.48.12 Constructor(RecordType as String, zoneID as CKRecordZoneIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.48.9 Constructor 395
- 6.48.10 Constructor(RecordType as String) 396
- 6.48.11 Constructor(RecordType as String, recordID as CKRecordIDMBS) 396
- 6.48.13 Constructor(rootRecord as CKRecordMBS) 397
- 6.48.14 Constructor(rootRecord as CKRecordMBS, shareID as CKRecordIDMBS) 397
- 6.48.15 Constructor(zoneID as CKRecordZoneIDMBS) 398

6.48.13 Constructor(rootRecord as CKRecordMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a share object.
Notes: When saving a newly created CKShare, you must save the share and its rootRecord in the same CKModifyRecordsOperationMBS batch.
See also:

- 6.48.9 Constructor 395
- 6.48.10 Constructor(RecordType as String) 396
- 6.48.11 Constructor(RecordType as String, recordID as CKRecordIDMBS) 396
- 6.48.12 Constructor(RecordType as String, zoneID as CKRecordZoneIDMBS) 397
- 6.48.14 Constructor(rootRecord as CKRecordMBS, shareID as CKRecordIDMBS) 397
- 6.48.15 Constructor(zoneID as CKRecordZoneIDMBS) 398

6.48.14 Constructor(rootRecord as CKRecordMBS, shareID as CKRecordIDMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a share object.
Notes: rootRecord: The record being shared.
shareID: The CKRecordID associated with this share.

When saving a newly created CKShare, save the share and its rootRecord in the same CKModifyRecordsOperationMBS batch.
See also:

6.48.15 Constructor(zoneID as CKRecordZoneIDMBS)

Plugin Version: 21.3, Platform: macOS, Targets: All.

Function: Creates a new share for the specified record zone.
Notes: ZoneID: The ID of the record zone to share.

A shared record zone must have the CKRecordZoneCapabilityZoneWideSharing capability. Custom record zones that you create in the user,Äôs private database have this capability by default. A record zone, and the records it contains, can take part in only a single share.

After accepting a share invite, CloudKit adds the records of the shared record zone to a new zone in the participant,Äôs shared database. Use CKFetchDatabaseChangesOperationMBS to fetch the ID of the new record zone. Then configure CKFetchRecordZoneChangesOperationMBS with that record zone ID and execute the operation to fetch the records.
If you use CKFetchShareMetadataOperationMBS to fetch the metadata for a shared record zone, the operation ignores the shouldFetchRootRecord and rootRecordDesiredKeys properties because, unlike a shared record hierarchy, a record zone doesn,Äôt have a nominated root record.

Available in macOS 12 or later.
See also:

6.48.16 participants as CKShareParticipantMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An array containing metadata objects for all participants in this share.
Notes: The participants array contains all participants on the share that the current user has permissions to see. At a minimum, this array includes the owner and the current user.

6.48.17 removeParticipant(participant as CKShareParticipantMBS)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Removes a participant from the share.
Notes: participant: The participant to be removed from this share.

To modify the list of participants, a share must have publicPermission set to CKShareParticipantPermission-None. You cannot mix and match private users and public users in the same share. Only certain participant types may be added via this API; see CKShareParticipantMBS.

6.48.18 Properties

6.48.19 currentUserParticipant as CKShareParticipantMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The current user,Äôs metadata for this share.
Notes: This computed property accesses the participants property and returns the CKShareParticipant that is associated with the current user.
(Read only property)

6.48.20 owner as CKShareParticipantMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The share owner,Äôs metadata.
Notes: This computed property accesses the participants property and returns the CKShareParticipant that is associated with the owner of this share.
(Read only property)

6.48.21 publicPermission as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Definition of permissions new users joining this share will have initially.
Notes: Shares with publicPermission more permissive than CKShareParticipantPermissionNone are joinable by any user with access to the share,Äôs URL. If the value is set to CKShareParticipantPermissionReadWrite, then those users can also create and modify records in the share. By default, public permission is CKShareParticipantPermissionNone. Changing the public permission to CKShareParticipantPermissionNone results in all public participants being removed when the share is saved.
(Read and Write property)

6.48.22 URL as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A URL that can be used to invite participants to this share.
Notes: This property is only available after a share record has been saved to the server. This URL is stable and is tied to the rootRecord. If you share a rootRecord, delete the share, and then re-share the same rootRecord via a newly created share, that newly created share,Äôs URL will be identical to the prior share,Äôs URL.
(Read only property)

6.49 class CKShareMetadataMBS

6.49.1 class CKShareMetadataMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to the metadata for a shared record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 21.3pr1

6.49.2 Methods

6.49.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.49.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.49.5 copy as CKShareMetadataMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.49.6 Properties

6.49.7 containerIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique identifier for the container associated with this share.
Notes: This read only property can be used to fetch the container associated with this share.
(Read only property)

6.49.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.49.9 ownerIdentity as CKUserIdentityMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The identity of the owner of the associated share.
Notes: This read only property is used to access the owner of the share.
(Read only property)

6.49.10 participantPermission as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Permissions associated with the user who retrieved the metadata for this share.
Notes: This property reflects the permissions associated with the participant who invoked the CKFetchShareMetadataOperationMBS.
(Read only property)

6.49.11 participantStatus as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The participation status of the user who retrieved the metadata for this share.
Notes: This property reflects the status of the participant who invoked the CKFetchShareMetadataOperationMBS. See CKShareParticipantMBS for the statuses available.
(Read only property)

6.49.12 participantType as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of the user who retrieved the metadata for this share.
Notes: This property reflects the participantType of the participant who invoked the CKFetchShareMetadataOperationMBS. See CKShareParticipantMBS for the types available.
(Read only property)

6.49.13 rootRecord as CKRecordMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The root record associated with this share.
Notes: This read only property contains a reference to the root record associated with this share. This property can be nil if the CKFetchShareMetadataOperationMBS did not set shouldFetchRootRecord to true.
(Read only property)

6.49.14 rootRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique identifier for the root record associated with this share.
Notes: This property is populated even if shouldFetchRootRecord is set to false on the CKFetchShareMetadataOperationMBS and can be used to retrieve the rootRecord if the record was not retrieved with the metadata.
(Read only property)

6.49.15 share as CKShareMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The associated share.
Notes: This read only property contains a reference to the CKShareMBS that this CKShareMetadataMBS is associated with.
(Read only property)

6.50 class CKShareParticipantMBS

6.50.1 class CKShareParticipantMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to a person who accepted a shared record..
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 21.5pr7

6.50.2 Methods

6.50.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.50.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.50.5 copy as CKShareParticipantMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.50.6 Properties

6.50.7 acceptanceStatus as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The current state of the user,Äôs acceptance of this share. (read-only)
Notes: This property contains the current state of the participant,Äôs acceptance of this share. For a list of possible values, see CKShareParticipantAcceptanceStatus* constants.
(Read only property)

6.50.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.50.9 permission as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The permission level that the user has for this share.
Notes: This property controls the permissions that the participant has for this share. For a list of possible values, see CKShareParticipantPermission* constants.
(Read and Write property)

6.50.10 role as Integer

Plugin Version: 21.5, Platform: macOS, Targets: All.

Function: The participant,Äôs role for the share.
Notes: The property controls the participant role for the share.
For a list of possible values, see CKShareParticipantRole* constants.
(Read and Write property)

6.50.11 type as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use role instead. **Function:** The participant type.
Notes: The property controls the participant type for the share.
For a list of possible values, see CKShareParticipantType* constants.
(Read and Write property)

6.50.12 userIdentity as CKUserIdentityMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The identity of the participant. (read-only)
Notes: This property contains a reference to the user identity for the share participant.
(Read only property)

6.50.13 Constants

Acceptance Status

Constant	Value	Description
CKShareParticipantAcceptanceStatusAccepted	2	The participant has accepted the share request.
CKShareParticipantAcceptanceStatusPending	1	The participant has not accepted the share request.
CKShareParticipantAcceptanceStatusRemoved	3	The participant was removed from the share.
CKShareParticipantAcceptanceStatusUnknown	0	The participant,Äôs status is unknown.

Permissions

Constant	Value	Description
CKShareParticipantPermissionNone	1	The participant does not have any permissions for this share.
CKShareParticipantPermissionReadOnly	2	The participant has read only permissions for this share.
CKShareParticipantPermissionReadWrite	3	The participant has full read and write permissions for this share.
CKShareParticipantPermissionUnknown	0	The current status of the participant,Äôs permissions is unknown.

Participant Roles.

Constant	Value	Description
CKShareParticipantRoleOwner	1	The participant is the owner of the share. As owner, the user can add pri users to the share.
CKShareParticipantRolePrivateUser	3	The participant is a private user and can access the share.
CKShareParticipantRolePublicUser	4	The participant is a private user and can access the share.
CKShareParticipantRoleUnknown	0	The type of the participant cannot be determined.

Participant Types.

Constant	Value	Description
CKShareParticipantTypeOwner	1	The participant is the owner of the share. As owner, the user can add pri users to the share.
CKShareParticipantTypePrivateUser	3	The participant is a private user and can access the share.
CKShareParticipantTypePublicUser	4	The participant is a public user. Public users are ,Äúself-added,Äù when participant accesses the shareURL. Owners cannot add public users.
CKShareParticipantTypeUnknown	0	The type of the participant cannot be determined.

6.51 class CKSubscriptionMBS

6.51.1 class CKSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Use a CKSubscription object to track changes occurring on the server.
Notes: A subscription acts like a persistent query on the server that can track the creation, deletion, and modification of records. When changes occur, they trigger the delivery of push notifications so that your app can respond appropriately.

see
https://developer.apple.com/reference/cloudkit/cksubscription

Available in 10.10 in 64-bit, but most is deprecated by Apple with 10.12.

6.51.2 Methods

6.51.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.10 and newer in 64-bit application.

6.51.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

6.51.5 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a query-based subscription that monitors records with the specified options.
Notes: recordType: The string that identifies the type of records to track. You are responsible for naming your app,Äôs record types. This parameter must not be nil or an empty string.
predicate: The matching criteria to apply to the records. This parameter must not be nil. For information about the operators that are supported in search predicates, see the discussion in CKQueryMBS.
subscriptionOptions: A bitmask of the configuration options for the subscription. You must specify at least one of the following values: CKSubscriptionOptionsFiresOnRecordCreation, CKSubscriptionOptionsFiresOnRecordUpdate, or CKSubscriptionOptionsFiresOnRecordDeletion.

Returns a subscription object initialized to track record-related changes.

The object returned by this method is configured as a query-based subscription for searching records in the target database. The subscription monitors the specified type of records in all of the user,Äôs record zones and generates push notifications when the search criteria are met.
See also:

- 6.51.4 Constructor 407
- 6.51.6 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer) 408
- 6.51.7 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string, subscriptionOptions as Integer) 409
- 6.51.8 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionOptions as Integer) 410

6.51.6 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a query-based subscription that monitors records with the specified options.
Notes: recordType: The string that identifies the type of records to track. You are responsible for naming your app,Äôs record types. This parameter must not be nil or an empty string.
predicate: The matching criteria to apply to the records. This parameter must not be nil. For information about the operators that are supported in search predicates, see the discussion in CKQueryMBS.
subscriptionOptions: A bitmask of the configuration options for the subscription. You must specify at least one of the following values: CKSubscriptionOptionsFiresOnRecordCreation, CKSubscriptionOptionsFiresOnRecordUpdate, or CKSubscriptionOptionsFiresOnRecordDeletion.

Returns a subscription object initialized to track record-related changes.

The object returned by this method is configured as a query-based subscription for searching records in the target database. The subscription monitors the specified type of records in all of the user,Äôs record zones and generates push notifications when the search criteria are met.
See also:

- 6.51.4 Constructor 407
- 6.51.5 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer) 408
- 6.51.7 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string, subscriptionOptions as Integer) 409
- 6.51.8 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionOptions as Integer) 410

6.51.7 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string, subscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a subscription object that monitors the specified zone and has a custom name that you provide.
Notes: zoneID: The ID of the record zone containing the records you want to monitor. This parameter must not be nil.
subscriptionID: The unique name of the subscription object. This string must be unique for all other subscription objects in the container. This parameter must not be nil.
subscriptionOptions: The configuration options for the subscription. You must specify 0 for this parameter. Zone subscriptions currently do not support any options.

Returns a subscription object initialized to track changes to the contents of a record zone.

The object returned by this method is configured as a zone-based subscription, which generates a push notification when any changes are made to the records in the specified zone.
See also:

- 6.51.4 Constructor 407
- 6.51.5 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer) 408
- 6.51.6 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer) 408
- 6.51.8 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionOptions as Integer) 410

6.51.8 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionOptions as Integer)

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a subscription object that monitors all records in the specified record zone.
Notes: zoneID: The ID of the record zone containing the records you want to monitor. This parameter must not be nil.
subscriptionOptions: The configuration options for the subscription. You must specify 0 for this parameter. Zone subscriptions currently do not support any options.

Returns a subscription object initialized to track changes to the contents of a record zone.

The object returned by this method is configured as a zone-based subscription, which generates a push notification when any changes are made to the records in the specified zone.
See also:

- 6.51.4 Constructor 407
- 6.51.5 Constructor(RecordType as String, predicate as NSPredicateMBS, querySubscriptionOptions as Integer) 408
- 6.51.6 Constructor(RecordType as String, predicate as NSPredicateMBS, subscriptionID as string, querySubscriptionOptions as Integer) 408
- 6.51.7 Constructor(zoneID as CKRecordZoneIDMBS, subscriptionID as string, subscriptionOptions as Integer) 409

6.51.9 copy as CKSubscriptionMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.51.10 Properties

6.51.11 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.51.12 notificationInfo as CKNotificationInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The configuration data for push notifications sent by the subscription.
Notes: If you want your subscription,Äôs push notifications to alert the user to corresponding changes, assign a value to this property. The server uses the information in the CKNotificationInfoMBS object to determine the delivery options for notifications. For example, you can specify the alert text to display and the name of a special sound file to play. When a push notification involves a record, you can also specify which fields of the record to include in the push notification,Äôs payload data.

If you do not assign a value to this property, the server still sends push notifications to your app but those notifications do not cause the system to alert the user. The default value of this property is nil.
(Read only property)

6.51.13 predicate as NSPredicateMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The matching criteria to apply to records.
Notes: A query-based subscription uses its search predicate to identify potential matches for records. It combines the predicate information with the value in the subscriptionOptions property to determine the conditions under which to send a push notification to the app.

The search predicate defines the records that the subscription object monitors for changes. The value in this property is used only if the subscriptionTypeMBS property is set to CKSubscriptionTypeQuery; otherwise, it is ignored.
(Read only property)

6.51.14 recordType as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The record type being monitored in a query-based subscription.
Notes: The value of this property applies only to query-based subscriptions and is set automatically by the constructors. For all other types of subscription objects, the value of this property is ignored and set to nil.
(Read only property)

6.51.15 subscriptionID as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The unique identifier for the subscription.
Notes: If you initialize the subscription object using the constructor, the default value of this property is set to the value provided by those methods. In all other cases, an ID based on a UUID is generated for you automatically.
(Read only property)

6.51.16 subscriptionOptions as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The options for triggering notifications.
Notes: Set the value of this property at initialization time. When configuring a query-based subscription, you must specify at least one of the following values: CKSubscriptionOptionsFiresOnRecordCreation, CKSubscriptionOptionsFiresOnRecordUpdate, or CKSubscriptionOptionsFiresOnRecordDeletion.
(Read only property)

6.51.17 subscriptionType as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The type of behavior provided by the subscription.
Notes: The value of this property is set automatically at creation time. Use it to distinguish between query-based subscriptions and those that monitor specific types of data.
(Read only property)

6.51.18 zoneID as CKRecordZoneIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the record zone to monitor.
Notes: The value of this property applies both to query-based subscriptions and zone-based subscriptions. Specifying a record zone limits the search scope to the records in that zone. In the case of a zone-based subscription, the search encompasses all records in the zone. For a query-based subscription, the search encompasses only records of a specific type in that zone.

For zone-based subscriptions, the value of this property is set automatically by the constructor. For all other subscription types, the default value is nil. To apply a zone to a query-based subscriptions, you must assign a value explicitly.
(Read only property)

6.51.19 Constants

Configuration Options

Constant	Value	Description
OptionsFiresOnce	8	An option for sending a push notification only one time. After sending the push notification, the server deletes the subscription object. This option applies only to query-based subscriptions.
OptionsFiresOnRecordCreation	1	An option for generating a push notification when a record that matches the search criteria in the predicate property is created. This option applies only to query-based subscriptions subscriptions.
OptionsFiresOnRecordDeletion	4	An option for generating a push notification when a record that matched the search criteria in the predicate property was deleted. This option applies only to query-based subscriptions.
OptionsFiresOnRecordUpdate	2	An option for generating a push notification when changes are made to the fields of a record that cause the record to match the search criteria specified in the predicate property. This option applies only to query-based subscriptions.

Subscription Types

Constant	Value	Description
TypeDatabase	3	A constant indicating the subscription is a query-based subscription. This type of subscription tracks the creation, modification, or deletion of a specific type of record.
TypeQuery	1	A constant indicating the subscription is a query-based subscription. This type of subscription tracks the creation, modification, or deletion of a specific type of record.
TypeRecordZone	2	A constant indicating the subscription is a zone-based subscription. The subscription tracks changes to records in a specific record zone.

6.52 class CKUserIdentityLookupInfoMBS

6.52.1 class CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An object that represents information you use to fetch users.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.52.2 Methods

6.52.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.52.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.
Notes: Please use the shared methods to constructor instances.
This constructor is private to make sure you don't create an object from this class by error. Please use designated functions to create objects.

6.52.5 copy as CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.52.6 lookupInfosWithEmailAddress(emailAddress as string) as CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of user identity lookup objects configured with email addresses as the search criteria.

Notes: emails: An array of strings that represent the email addresses to use to configure the CKUserIdentityLookupInfoMBS objects.

Returns an array of CKUserIdentityLookupInfoMBS objects that are configured with the email addresses that were passed in.
May return nil in case of error.

Once initialized, these objects can be passed into a CKDiscoverUserIdentitiesOperationMBS, or a CKFetchShareParticipantsOperationMBS to retrieve the CKUserIdentity objects for the user.

6.52.7 lookupInfosWithEmails(emailAddresses() as string) as CKUserIdentityLookupInfoMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of user identity lookup objects configured with email addresses as the search criteria.
Notes: emails: An array of strings that represent the email addresses to use to configure the CKUserIdentityLookupInfoMBS objects.

Returns an array of CKUserIdentityLookupInfoMBS objects that are configured with the email addresses that were passed in.
May return nil in case of error.

Once initialized, these objects can be passed into a CKDiscoverUserIdentitiesOperationMBS, or a CKFetchShareParticipantsOperationMBS to retrieve the CKUserIdentityMBS objects for the user.

6.52.8 lookupInfosWithPhoneNumbers(phoneNumbers() as string) as CKUserIdentityLookupInfoMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of user identity lookup objects configured with phone numbers as the search criteria.
Notes: phoneNumbers: An array of strings that represent the phone numbers to use to configure the CKUserIdentityLookupInfoMBS objects.

Returns an array of CKUserIdentityLookupInfoMBS objects that are configured with the phone numbers that were passed in.
May return nil in case of error.

Once initialized, these objects can be passed into a CKDiscoverUserIdentitiesOperationMBS, or a CKFetchShareParticipantsOperationMBS to retrieve the CKUserIdentityMBS objects for the user.

6.52.9 lookupInfosWithRecordIDs(userRecordIDs() as CKRecordIDMBS) as CKUserIdentityLookupInfoMBS()

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Returns an array of user identity lookup objects configured with phone numbers as the search criteria.
Notes: userRecordIDs: An array of CKRecordIDMBS objects that are used to configure the CKUserIdentityLookupInfoMBS objects.

Returns an array of CKUserIdentityLookupInfoMBS objects that are configured with the CKRecordIDMBS objects that were passed in.one

6.52.10 lookupInfosWithUserRecordID(userRecordID as CKRecordIDMBS) as CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a look up info object with the user record ID property configured.
Notes: userRecordID: The user record ID to use to look up the user.

Returns an user identity lookup object initialized to look up a user by the user,Äôs record ID, or nil if the object cannot be initialized.

Once initialized, this object can be passed into CKDiscoverUserIdentitiesOperationMBS, or a CKFetchShareParticipantsOperationMBS to retrieve the CKUserIdentityMBS for the user.

6.52.11 lookupInfosWithWithPhoneNumber(phoneNumber as string) as CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Initializes and returns a look up info object with the phone number property configured.
Notes: phoneNumber: The phone number to use to look up the user.

Returns an user identity lookup object initialized to look up a user by the user,Äôs phone number, or nil if the object cannot be initialized.

Once initialized, this object can be passed into a CKDiscoverUserIdentitiesOperationMBS, or a CKFetchShareParticipantsOperationMBS to retrieve the CKUserIdentityMBS for the user.

6.52.12 Properties

6.52.13 emailAddress as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The email address of the user whose information you want to retrieve.
Notes: This property corresponds to a single email address associated with the user whose information you wish to look up.
(Read only property)

6.52.14 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.52.15 phoneNumber as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The phone number of the user whose information you want to retrieve.
Notes: This property corresponds to a single phone number associated with the user whose information you wish to look up.
(Read only property)

6.52.16 userRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The ID of the user record.
Notes: Use this value to retrieve the user record associated with the specified user. The user record does not contain any personal information about the user by default. Your app can add data to the user record but should not add any sensitive user data to it.
(Read only property)

6.53 class CKUserIdentityMBS

6.53.1 class CKUserIdentityMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A reference to a user.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.53.2 Methods

6.53.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.12 and newer in 64-bit application.

6.53.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.53.5 copy as CKUserIdentityMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.53.6 Properties

6.53.7 description as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The description text.
Notes: (Read only property)

6.53.8 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.53.9 hasiCloudAccount as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether this user has an iCloud account.
Notes: The value is true if this CKUserIdentity has an iCloud account associated with it; otherwise, false.
(Read only property)

6.53.10 localizedDisplayName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The localized display name.
Notes: (Read only property)

6.53.11 lookupInfo as CKUserIdentityLookupInfoMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The information (phone number, email address, etc.) used to retrieve this user.
Notes: The information that is passed into a CKDiscoverUserIdentitiesOperationMBS or CKFetchShareParticipantsOperationMBS object to retrieve the user identity.
(Read only property)

6.53.12 nameComponents as NSPersonNameComponentsMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The name of the user associated with the specified user identity object.
Notes: This property can be used to display the name of the user associated with this CKUserIdentity object.
(Read only property)

6.53.13 userRecordID as CKRecordIDMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Unique identifier for this record.
Notes: This property is the unique identifier associated with this CKUserIdentity.
(Read only property)

6.54 class NSPersonNameComponentsMBS

6.54.1 class NSPersonNameComponentsMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: An NSPersonNameComponents object encapsulates the components of a person's name in an extendable, object-oriented manner.
Notes: It is used to specify a person's name by providing the components comprising a full name: given name, middle name, family name, prefix, suffix, nickname, and phonetic representation.
Blog Entries

- MBS Xojo Plugins, version 23.2pr5
- News from the MBS Xojo Plugins in version 21.5
- MBS Xojo Plugins, version 21.5pr7

6.54.2 Methods

6.54.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true for OS X 10.11 and newer.

6.54.4 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The constructor.

6.54.5 copy as NSPersonNameComponentsMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Creates a copy of this object.

6.54.6 formatted as NSAttributedStringMBS

Plugin Version: 21.5, Platform: macOS, Targets: All.

Function: Provides a formatted representation of the name.
Example:

```
Dim n As New NSPersonNameComponentsMBS

n.familyName = "Smith"
n.givenName = "Bob"

Dim a As NSAttributedStringMBS = n.formatted

Dim tv As NSTextViewMBS = TextArea1.NSTextViewMBS
tv.textStorage.setAttributedString a
```

6.54.7 Properties

6.54.8 familyName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Name bestowed upon an individual to denote membership in a group or family. (for example, ,ÄúAppleseed,Äù).
Notes: (Read and Write property)

6.54.9 givenName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Name bestowed upon an individual to differentiate them from other members of a group that share a family name (for example, ,ÄúJohnathan,Äù).
Notes: (Read and Write property)

6.54.10 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.54.11 middleName as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Secondary name bestowed upon an individual to differentiate them from others that have the same given name (for example, ,ÄúMaple,Äù).
Notes: (Read and Write property)

6.54.12 namePrefix as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The portion of a name,Äôs full form of address that precedes the name itself (for example, ,ÄúDr.,,Äù ,ÄúMr.,,Äù ,ÄúMs.,Äù).
Notes: (Read and Write property)

6.54.13 nameSuffix as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The portion of a name,Äôs full form of address that follows the name itself (for example, ,ÄúEsq.,,Äù ,ÄúJr.,,Äù ,ÄúPh.D.,Äù).
Notes: (Read and Write property)

6.54.14 nickname as String

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: Name substituted for the purposes of familiarity (for example, "Johnny").
Notes: (Read and Write property)

6.54.15 phoneticRepresentation as NSPersonNameComponentsMBS

Plugin Version: 16.5, Platform: macOS, Targets: All.

Function: The phonetic representation name components of the receiver.
Notes: Each component of the receiver with a value should have a corresponding value for any value set for this property. nil by default.
(Read and Write property)

Chapter 7

Contacts

7.1 class CNChangeHistoryAddContactEventMBS

7.1.1 class CNChangeHistoryAddContactEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A contact was added.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.1.2 Methods

7.1.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.1.4 Properties

7.1.5 contact as CNContactMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The contact added.
Notes: (Read only property)

7.1.6 containerIdentifier as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The identifier for the container.
Notes: (Read only property)

7.2 class CNChangeHistoryAddGroupEventMBS

7.2.1 class CNChangeHistoryAddGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A group was added.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.2.2 Methods

7.2.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.2.4 Properties

7.2.5 containerIdentifier as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The identifier for the container.
Notes: (Read only property)

7.2.6 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group.
Notes: (Read only property)

7.3 class CNChangeHistoryAddMemberToGroupEventMBS

7.3.1 class CNChangeHistoryAddMemberToGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A contact was added to a group.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.3.2 Methods

7.3.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.3.4 Properties

7.3.5 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group.
Notes: (Read only property)

7.3.6 member as CNContactMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The member added.
Notes: (Read only property)

7.4 class CNChangeHistoryAddSubgroupToGroupEventMBS

7.4.1 class CNChangeHistoryAddSubgroupToGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A subgroup was added to a group.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.4.2 Methods

7.4.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.4.4 Properties

7.4.5 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group.
Notes: (Read only property)

7.4.6 subgroup as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The sub group.
Notes: (Read only property)

7.5 class CNChangeHistoryDeleteContactEventMBS

7.5.1 class CNChangeHistoryDeleteContactEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A contact was removed.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.5.2 Methods

7.5.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.5.4 Properties

7.5.5 contactIdentifier as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The identifier for the deleted contact.
Notes: (Read only property)

7.6 class CNChangeHistoryDeleteGroupEventMBS

7.6.1 class CNChangeHistoryDeleteGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A group was deleted.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.6.2 Methods

7.6.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.6.4 Properties

7.6.5 groupIdentifier as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The identifier on the deleted group.
Notes: (Read only property)

7.7 class CNChangeHistoryDropEverythingEventMBS

7.7.1 class CNChangeHistoryDropEverythingEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Drop all cached information your app has persisted.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.7.2 Methods

7.7.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.8 class CNChangeHistoryEventMBS

7.8.1 class CNChangeHistoryEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The base class for changing events.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.0
- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.8.2 Methods

7.8.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.8.4 Properties

7.8.5 className as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The class name of the Objective-C class.
Notes: (Read only property)

7.8.6 Handle as Integer

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

7.9 class CNChangeHistoryFetchRequestMBS

7.9.1 class CNChangeHistoryFetchRequestMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Specifies the criteria to fetch change history.
Notes: Changes to contacts are always returned.
All changes are coalesced to remove redundant adds, updates and deletes.
This request is used with enumeratorForChangeHistoryFetchRequest in CNContactStoreMBS class.
Subclass of the CNContactFetchRequestMBS class.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.9.2 Methods

7.9.3 additionalContactKeyDescriptors as CNKeyDescriptorMBS()

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Additional keys to include in the fetched contacts.

7.9.4 available as Boolean

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Returns true on macOS 10.15 or iOS 13 or later.

7.9.5 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The constructor.

7.9.6 excludedTransactionAuthors as String()

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Exclude changes made by certain authors.
Notes: If set, transactions made by the specified authors will be excluded from the results. Use this, in conjunction with CNSaveRequest.transactionAuthor, to suppress processing of changes you already know about.

7.9.7 setAdditionalContactKeyDescriptors(additionalContactKeyDescriptors() as CNKeyDescriptorMBS)

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Additional keys to include in the fetched contacts.
Notes: By default, only CNContactIdentifierKey will be fetched. If you would like to include additional key descriptors to process the contacts, include the key descriptors you need.

CNContactIdentifierKey will always be fetched, whether you request it or not.

7.9.8 setExcludedTransactionAuthors(setExcludedTransactionAuthors() as String)

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Exclude changes made by certain authors.
Notes: If set, transactions made by the specified authors will be excluded from the results. Use this, in conjunction with CNSaveRequest.transactionAuthor, to suppress processing of changes you already know about.

7.9.9 Properties

7.9.10 includeGroupChanges as Boolean

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Set to true to also fetch group changes.
Notes: Default is false.
(Read and Write property)

7.9.11 mutableObjects as Boolean

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: To return mutable contacts and groups.
Notes: If true returns mutable contacts and groups. Default is false.

(Read and Write property)

7.9.12 shouldUnifyResults as Boolean

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns contact changes as unified contacts.
Notes: If true, returns unified contact history. Otherwise returns individual contact history. Default is true.

A unified contact is the aggregation of properties from a set of linked individual contacts.
If an individual contact is not linked then the unified contact is simply that individual contact.
(Read and Write property)

7.9.13 startingToken as MemoryBlock

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Request changes made after a certain point.
Notes: If non-nil, only changes made after this point in history will be returned.

If nil, a CNChangeHistoryDropEverythingEventMBS will be returned, followed by an add event for every contact and group currently in the contacts database.
(Read and Write property)

7.10 class CNChangeHistoryRemoveMemberFromGroupEventMBS

7.10.1 class CNChangeHistoryRemoveMemberFromGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A contact was removed from a group.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.10.2 Methods

7.10.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.10.4 Properties

7.10.5 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group.
Notes: (Read only property)

7.10.6 member as CNContactMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The member removed.
Notes: (Read only property)

7.11 class CNChangeHistoryRemoveSubgroupFromGroupEventMBS

7.11.1 class CNChangeHistoryRemoveSubgroupFromGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A subgroup was removed from a group.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.11.2 Methods

7.11.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.11.4 Properties

7.11.5 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group.
Notes: (Read only property)

7.11.6 subgroup as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The sub group.
Notes: (Read only property)

7.12 class CNChangeHistoryUpdateContactEventMBS

7.12.1 class CNChangeHistoryUpdateContactEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A contact was updated.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.12.2 Methods

7.12.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.12.4 Properties

7.12.5 contact as CNContactMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The contact changed.
Notes: (Read only property)

7.13 class CNChangeHistoryUpdateGroupEventMBS

7.13.1 class CNChangeHistoryUpdateGroupEventMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A group was updated.
Notes: Subclass of the CNChangeHistoryEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Contacts history for Xojo
- MBS Xojo Plugins, version 22.0pr7

7.13.2 Methods

7.13.3 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.13.4 Properties

7.13.5 group as CNGroupMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The group used.
Notes: (Read only property)

7.14 class CNContactFetchRequestMBS

7.14.1 class CNContactFetchRequestMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A contact fetch request.
Notes: The CNContactFetchRequest class defines fetching options to use while fetching contacts. It is required to have contact property key(s) to fetch a contact,Äôs properties. Use this class with the enumerateContactsWithFetchRequest method to execute the contact fetch request.
Blog Entries

- MBS Xojo Plugins, version 18.4pr5

7.14.2 Methods

7.14.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.14.4 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The designated initializer for a fetch request that uses all keys.
Notes: Same as calling other Constructor and passing result of CNContactStoreMBS.AllFetchKeys function.
See also:

- 7.14.5 Constructor(keysToFetch() as CNKeyDescriptorMBS) 441

7.14.5 Constructor(keysToFetch() as CNKeyDescriptorMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The designated initializer for a fetch request that uses the specified keys.
Notes: keysToFetch: An array of contact property keys and/or key descriptors from contacts objects to be fetched in the returned contacts.
See also:

- 7.14.4 Constructor 441

7.14.6 keysToFetch as CNKeyDescriptorMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The properties to fetch in the returned contacts.
Notes: An array of contact property keys or key descriptors from contact objects to be fetched in the returned contacts. For example, CNContactEmailAddressesKey, CNContactPhoneNumbersKey, CNContactFormatterStyleFullName fetches the contact,Äôs email addresses, phone numbers, and contact,Äôs full name with the contact formatter.

7.14.7 setKeysToFetch(keysToFetch() as CNKeyDescriptorMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the properties to fetch in the returned contacts.

7.14.8 Properties

7.14.9 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactFetchRequest object.
(Read and Write property)

7.14.10 mutableObjects as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether to return mutable contacts.
Notes: When the value of this property is true, the fetch returns CNMutableContact objects; otherwise it returns CNContact objects. The default value of this property is false.
(Read and Write property)

7.14.11 predicate as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The predicate to match contacts against.
Notes: Set the value of this property to nil to match all contacts or use the predicates define in CNContact

Predicates. Compound predicates are not supported.
(Read and Write property)

7.14.12 SortOrder as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The sort order for contacts.
Notes: The default sort order is CNContactSortOrderNone.
(Read and Write property)

7.14.13 unifyResults as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether to return linked contacts as unified contacts.
Notes: A unified contact is an aggregation of properties from a set of linked individual contacts. When the value of this property is true, the fetch returns unified contacts; otherwise, it returns individual contacts. The default value of this property is true.
(Read and Write property)

7.15 class CNContactFormatterMBS

7.15.1 class CNContactFormatterMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactFormatter class defines the different formatting styles for contacts.
Example:

```
dim m as new CNMutableContactMBS
m.givenName = "Bob"
m.familyName = "Miller"

dim f as new CNContactFormatterMBS
dim style as Integer = CNContactFormatterMBS.CNContactFormatterStyleFullName
MsgBox CNContactFormatterMBS.stringFromContact(m, style)
```

Notes: This class handles international ordering and delimiting for the contact name components. When formatting many contacts, create an instance of this class and use the instance methods; otherwise use the class methods.

7.15.2 Methods

7.15.3 attributedStringFromContact(contact as CNContactMBS, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Formats the contact name as an attributed string.
Notes: contact: The contact whose name is to be formatted.
attributes: The default attributes to use. For more information, see NSFormatter.

Returns the formatted contact name as an attributed string.

This method behaves similarly to stringFromContact, except that it returns an attributed string. It includes the attribute key CNContactPropertyAttribute whose attribute values are contact property keys, such as CNContactGivenNameKey. This identifies the name components in the formatted contact name.
See also:

- 7.15.4 attributedStringFromContact(contact as CNContactMBS, Style as Integer, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS

7.15.4 attributedStringFromContact(contact as CNContactMBS, Style as Integer, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Formats the contact name as an attributed string.
Notes: contact: The contact whose name is to be formatted.
style: The formatting style to be used for the contact name.
attributes: The default attributes to use. For more information, see NSFormatter.

Returns the formatted contact name as an attributed string.

This method behaves similarly to stringFromContact, except that it returns an attributed string. It includes the attribute key CNContactPropertyAttribute, whose attribute values are contact property keys, such as CNContactGivenNameKey. This identifies the name components in the formatted contact name.
See also:

- 7.15.3 attributedStringFromContact(contact as CNContactMBS, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS 444

7.15.5 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.15.6 CNContactPropertyAttribute as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the attributes in the attribute dictionary.
Notes: If the attributes include the key CNContactPropertyAttribute whose attribute values are contact property keys, such as CNContactGivenNameKey. This identifies the name components in the formatted contact name.

7.15.7 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.15.8 delimiterForContact(contact as CNContactMBS) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the delimiter to use between name components.
Notes: contact: The contact whose name is to be formatted.

Returns the delimiter to use between name components.

If contact is nil, or if it has no first name, middle name, or last name, this method returns an empty string.

7.15.9 descriptorForRequiredKeysForStyle(Style as Integer) as CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the required key descriptor for the specified formatting style of the contact.
Notes: style: The formatting style to be used for contact name.

Returns the contact key descriptor for the formatting style.

Include this method with the keys to fetch when fetching contacts. To format multiple styles, you can include multiple key descriptors with the keys to fetch.

7.15.10 nameOrderForContact(contact as CNContactMBS) as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the display name order.
Notes: contact: The contact whose name is to be formatted.

Returns the display order to use when combining the given name and family name components.

For more information about display name orders, see CNContactDisplayNameOrder.

7.15.11 stringFromContact(contact as CNContactMBS) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Formats the contact name.
See also:

- 7.15.12 stringFromContact(contact as CNContactMBS, Style as Integer) as String 447

7.15.12 stringFromContact(contact as CNContactMBS, Style as Integer) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the contact name, formatted with the specified formatter.
Notes: contact: The contact whose name is to be formatted.
style: The formatting style to be used for the contact name.
See also:

- 7.15.11 stringFromContact(contact as CNContactMBS) as String 446

7.15.13 Properties

7.15.14 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactFormatter object.
(Read and Write property)

7.15.15 Style as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The formatting style for the contact name.
Notes: The style for a contact formatter instance. The default value for this property is CNContactFormatterStyleFullName. For more information on formatting styles, see CNContactFormatterStyle.
(Read and Write property)

7.15.16 Constants

Display Name Order

Constant	Value	Description
CNContactDisplayNameOrderFamilyNameFirst	2	Display name order by family name first.
CNContactDisplayNameOrderGivenNameFirst	1	Display name order by given name first.
CNContactDisplayNameOrderUserDefault	0	Display name order by user default.

Styles

Constant	Value	Description
CNContactFormatterStyleFullName	0	Combines the contact name components into a full name.
CNContactFormatterStylePhoneticFullName	1	Combines the contact phonetic name components into a phonetic

7.16 class CNContactMBS

7.16.1 class CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The class for a contact.
Notes: The CNContact is a thread-safe class that represents an immutable value object for contact properties, such as the first name and phone numbers of a contact. CNContact is similar to a complex Foundation collection, in that it has a mutable subclass (CNMutableContact). Neither the CNContact nor CNMutableContact class maintain a reference to their data store. Every contact has a unique ID, which you obtain using the identifier property.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- CNContactPickerViewController for Xojo
- Contacts history for Xojo
- News from the MBS Xojo Plugins Version 20.5
- MBS Xojo Plugins, version 20.5pr1
- MBS Xojo Plugins, version 18.1pr2
- MBS Xojo Plugins, version 17.2pr5

7.16.2 Methods

7.16.3 areKeysAvailable(keyDescriptors() as CNKeyDescriptorMBS) as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether all contact property values for the specified keys are fetched.
Notes: The isKeyAvailable or areKeysAvailable methods are used where you are not certain of the keys that when fetched. If this method returns false, refetch the contact using the contact identifier and the keys you want to fetch. Accessing a property that was not fetched will throw an CNContactPropertyNotFetchedExceptionName exception.

7.16.4 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.16.5 CNContactBirthdayKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Birthday.

7.16.6 CNContactDatesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Contact dates.

7.16.7 CNContactDepartmentNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Department name.

7.16.8 CNContactEmailAddressesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Email address.

7.16.9 CNContactFamilyNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Family name.

7.16.10 CNContactGivenNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Given name.

7.16.11 CNContactIdentifierKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: The identifier.

7.16.12 CNContactImageDataAvailableKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Image data availability.
Available in OS X 10.12 or newer.

7.16.13 CNContactImageDataKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Image data.

7.16.14 CNContactInstantMessageAddressesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Instant messages.

7.16.15 CNContactJobTitleKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Job title.

7.16.16 CNContactMiddleNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Middle name.

7.16.17 CNContactNamePrefixKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Name prefix.

7.16.18 CNContactNameSuffixKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Name suffix.

7.16.19 CNContactNicknameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Nickname.

7.16.20 CNContactNonGregorianBirthdayKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Non-Gregorian birthday.

7.16.21 CNContactNoteKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Note.

7.16.22 CNContactOrganizationNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Organization name.

7.16.23 CNContactPhoneNumbersKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Phone number.

7.16.24 CNContactPhoneticFamilyNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Phonetic family name.

7.16.25 CNContactPhoneticGivenNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Phonetic given name.

7.16.26 CNContactPhoneticMiddleNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Phonetic middle name.

7.16.27 CNContactPhoneticOrganizationNameKey as String

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic spelling of the contact's organization name.
Notes: Requires macOS 10.12 or newer.

7.16.28 CNContactPostalAddressesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Postal address.

7.16.29 CNContactPreviousFamilyNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Previous family name.

7.16.30 CNContactPropertyNotFetchedExceptionName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Exception thrown when an accessed property was not fetched.
Notes: The plugin throws a NSExceptionMBS where the name is this value in case a property is not available.

7.16.31 CNContactRelationsKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Contact relations.

7.16.32 CNContactSocialProfilesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Social profile.

7.16.33 CNContactThumbnailImageDataKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Thumbnail data.

7.16.34 CNContactTypeKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: Contact type.

7.16.35 CNContactUrlAddressesKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for contact properties.
Notes: URL Address.

7.16.36 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

7.16.37 contactRelations as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled relations for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNContactRelation value. This property was previously known as related names.

7.16.38 copy as CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the contact object.

7.16.39 dates as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array containing labeled Gregorian dates.
Notes: This property is an array of CNLabeledValue objects, each of which has an NSString label and NSDateComponents value. You can use this property to store Gregorian dates such as anniversaries. Day and month components are required and year is optional. The calendar component can be nil or NSCalendarIdentifierGregorian. All other date components are invalid and including them results in an NSError object that includes the key paths of the invalid components and the error code CNErrorCodeValidationConfigurationError.

7.16.40 descriptorForAllComparatorKeys as CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all the keys required for the contact sort comparator.
Notes: This method implements the CNKeyDescriptor protocol and can be used as an array element when fetching keys for contacts.

7.16.41 emailAddresses as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled email addresses for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and an NSString that contains the email address.

7.16.42 instantMessageAddresses as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled IM addresses for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNInstantMessageAddress value.

7.16.43 isKeyAvailable(key as String) as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether the contact property value for the specified key is fetched.
Notes: The isKeyAvailable or areKeysAvailable methods are used when you are not certain of the keys that were fetched. If this method returns false, refetch the contact using the contact identifier and the keys you want to fetch. Accessing a property that was not fetched will throw CNContactPropertyNotFetchedExceptionName.

7.16.44 isUnifiedWithContactWithIdentifier(contactIdentifier as String) as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns true if the receiver was fetched as a unified contact and includes the contact having contactIdentifier in its unification.

7.16.45 localizedStringForKey(key as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a string containing the localized contact property name.
Notes: key: A string containing the contact property key.

Returns a localized string containing the contact property name.

This method returns a localized string for a contact property key. For example, the value of a Canadian CNContactPostalAddressesKey field would be ,ÄúPostal Code,Äù, while the value of a French one would be ,ÄúCode Postal,Äù.

7.16.46 mutableCopy as CNMutableContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a mutable copy of this contact.

7.16.47 phoneNumbers as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled phone numbers for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNPhoneNumber value.

7.16.48 postalAddresses as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled postal addresses for a contact.
Example:

```
Dim c As CNContactMBS // your contact

Dim postalAddresses() As CNLabeledValueMBS = c.postalAddresses
For Each postalAddresse As CNLabeledValueMBS In postalAddresses

Dim p As CNPostalAddressMBS = postalAddresse.Value

MsgBox p.Street + EndOfLine + p.PostalCode + " " + p.City
Next
```

Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNPostalAddress value.

7.16.49 predicateForContactsInContainerWithIdentifier(containerIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts in the specified container.
Notes: Returns a predicate that can be used to fetch contacts from CNContactStore.

7.16.50 predicateForContactsInGroupWithIdentifier(groupIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts that are members in the specified group.
Notes: Returns a predicate that can be used to fetch contacts from CNContactStore.

7.16.51 predicateForContactsMatchingEmailAddress(emailAddress as String) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts whose email address matches the specified value.
Example:

```
Dim keysToFetch() As CNKeyDescriptorMBS
keysToFetch.append CNContactVCardSerializationMBS.descriptorForRequiredKeys

Dim predicate As NSPredicateMBS = CNContactMBS.predicateForContactsMatchingEmailAddress("test@test.test")

Dim error As NSErrorMBS
Dim contacts() As CNContactMBS = m.unifiedContactsMatchingPredicate(predicate, keysToFetch, error)

If error <>Nil Then
List.AddRow "Error: "+error.LocalizedDescription
Else
List.AddRow Str(contacts.Ubound+1)+" contacts found"
End If
```

Notes: emailAddress: The email address to be matched.

Returns a predicate that you can use to fetch contacts from CNContactStoreMBS class.

Requires macOS 10.13 or newer.

7.16.52 predicateForContactsMatchingName(name as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts matching the specified name.
Example:

```
Dim keysToFetch() As CNKeyDescriptorMBS
keysToFetch.append CNContactVCardSerializationMBS.descriptorForRequiredKeys

Dim predicate As NSPredicateMBS = CNContactMBS.predicateForContactsMatchingName("Peter")

Dim error As NSErrorMBS
Dim contacts() As CNContactMBS = m.unifiedContactsMatchingPredicate(predicate, keysToFetch, error)

If error <>Nil Then
```

```
List.AddRow "Error: "+error.LocalizedDescription
Else
List.AddRow Str(contacts.Ubound+1)+" contacts found"
End If
```

Notes: The name can contain any number of words.
Returns a predicate that can be used to fetch contacts from CNContactStore.

7.16.53 predicateForContactsMatchingPhoneNumber(phoneNumber as CNPhoneNumberMBS) as NSPredicateMBS

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts whose phone number matches the specified value.
Example:

```
Dim keysToFetch() As CNKeyDescriptorMBS
keysToFetch.append CNContactVCardSerializationMBS.descriptorForRequiredKeys

Dim phone As New CNPhoneNumberMBS("123456789,Ä¨")
Dim predicate As NSPredicateMBS = CNContactMBS.predicateForContactsMatchingPhoneNumber(phone)

Dim error As NSErrorMBS
Dim contacts() As CNContactMBS = m.unifiedContactsMatchingPredicate(predicate, keysToFetch, error)

If error <>Nil Then
List.AddRow "Error: "+error.LocalizedDescription
Else
List.AddRow Str(contacts.Ubound+1)+" contacts found"
End If
```

Notes: Returns a predicate that you can use to fetch contacts from CNContactStoreMBS.

Requires macOS 10.13 or newer.

7.16.54 predicateForContactsWithIdentifiers(Identifiers() as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the contacts matching the specified identifiers.

Notes: Returns a predicate that can be used to fetch contacts from CNContactStore.

7.16.55 socialProfiles as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled social profiles for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNSocialProfile value.

7.16.56 urlAddresses as CNLabeledValueMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of labeled URL addresses for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and an NSString value that contains the URL.

7.16.57 valueForKey(key as String) as Variant

Plugin Version: 17.2, Platform: macOS, Targets: Desktop & iOS.

Function: Queries value for a given key.
Example:

```
// your contact
dim c as CNContactMBS

dim value as Variant = c.valueForKey(c.CNContactPreviousFamilyNameKey)
if value = nil then
// empty
else
MsgBox value.StringValue
end if
```

Notes: Normally you use the properties, but if you loop over a list of keys, you can use this function to query value for key.

7.16.58 Properties

7.16.59 birthday as NSDateComponentsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A date component for the Gregorian birthday of the contact.
Notes: Birthdays are represented by this property, whose values are the relevant properties of an NSDateComponents object. Day and month components are required for this property, and year is optional. The calendar component can be nil or NSCalendarIdentifierGregorian. All other date components are invalid and including them results in an NSError object that includes the key paths of the invalid components and the error code CNErrorCodeValidationConfigurationError.
(Read only property)

7.16.60 contactType as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An enum identifying the contact type.
Notes: Can be CNContactMBS.CNContactTypeOrganization or CNContactMBS.CNContactTypePerson.
(Read only property)

7.16.61 departmentName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the department associated with the contact.
Notes: (Read only property)

7.16.62 familyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The family name of the contact.
Notes: (Read only property)

7.16.63 givenName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The given name of the contact.

Notes: The given name is often known as the first name of the contact.
(Read only property)

7.16.64 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContact object.
(Read and Write property)

7.16.65 identifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A value that uniquely identifies a contact on the device.
Notes: It is recommended that you use the identifier when re-fetching the contact. An identifier can be persisted between the app launches. Note that this identifier only uniquely identifies the contact on the current device.
(Read only property)

7.16.66 imageData as MemoryBlock

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The profile picture of a contact.
Notes: It is recommended that you fetch this property only when you need to access its value, such as when you need to display the contact,Äôs profile picture.
(Read only property)

7.16.67 imageDataAvailable as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether a contact has a profile picture.
Notes: (Read only property)

7.16.68 jobTitle as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The contact,Äôs job title.
Notes: (Read only property)

7.16.69 middleName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The middle name of the contact.
Notes: (Read only property)

7.16.70 namePrefix as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name prefix of the contact.
Notes: (Read only property)

7.16.71 nameSuffix as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name suffix of the contact.
Notes: (Read only property)

7.16.72 nickname as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The nickname of the contact.
Notes: (Read only property)

7.16.73 nonGregorianBirthday as NSDateComponentsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A date component for the non-Gregorian birthday of the contact.
Notes: Non-Gregorian birthdays can be displayed using this property, whose values are the relevant properties of an NSDateComponents object. Day and month components are required; year and leap month are optional. The calendar component is also required and must be an NSCalendar object with an identifier other than NSCalendarIdentifierGregorian. For example, some supported calendars are Chinese, Hebrew, and Islamic. All other date components are invalid and including them results in an NSError object that includes the key paths of the invalid components and the error code CNErrorCodeValidationConfigurationError.
(Read only property)

7.16.74 note as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A string containing notes for the contact.
Notes: To fetch the note property in iOS 13 or later or macOS 16 or later, add the com.apple.developer.contacts.notes entitlement to your app. The entitlement requires permission from Apple to use, and you can,Äôt publicly distribute your app until you have permission to use it. For more information about adding the entitlement and getting permission, see com.apple.developer.contacts.notes.
(Read only property)

7.16.75 organizationName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the organization associated with the contact.
Notes: (Read only property)

7.16.76 phoneticFamilyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A string for the phonetic family name of the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs family (or last) name.
(Read only property)

7.16.77 phoneticGivenName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic given name of the contact.

Notes: This property contains a string that specifies the pronunciation of the contact,Äôs given (or first) name.
(Read only property)

7.16.78 phoneticMiddleName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic middle name of the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs middle name.
(Read only property)

7.16.79 phoneticOrganizationName as String

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic name of the organization associated with the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs organization name.
Requires macOS 10.12 or newer.
(Read only property)

7.16.80 previousFamilyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A string for the previous family name of the contact.
Notes: The previous family name is often known as the maiden name of the contact.
(Read only property)

7.16.81 thumbnailImageData as MemoryBlock

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The thumbnail version of the contact,Äôs profile picture.
Notes: The thumbnailImageData property is derived from the imageData property, including cropping information from vCards or edits from contact viewing. It is recommended that you fetch this property only when you need to access its value, such as when you need to display the contact,Äôs profile thumbnail picture.
(Read only property)

7.16.82 Constants

Sort Orders

Constant	Value	Description
CNContactSortOrderFamilyName	3	Order by Family Name.
CNContactSortOrderGivenName	2	Order by Given Name.
CNContactSortOrderNone	0	Order by no order.
CNContactSortOrderUserDefault	1	Order by user preference.

Contact Types

Constant	Value	Description
CNContactTypeOrganization	1	The contact is an Organization.
CNContactTypePerson	0	The contact is a person.

7.17 class CNContactPickerMBS

7.17.1 class CNContactPickerMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The CNContactPicker object displays the popover-based system interface for selecting a contact.
Notes: The methods and properties of this class help you choose a contact or a contact's value, such as a phone number or email address, of a contact.

7.17.2 Methods

7.17.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.17.4 close

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Closes the popover.

7.17.5 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The constructor.

7.17.6 Destructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The destructor.

7.17.7 displayedKeys as String()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The keys to be displayed when a contact is expanded.
Notes: If no keys are provided, the picker selects contacts instead of values. For a list of possible keys, see Metadata Keys in CNContact.

7.17.8 setDisplayedKeys(keys() as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Set the keys to be displayed when a contact is expanded.
Notes: If no keys are provided, the picker selects contacts instead of values. For a list of possible keys, see Metadata Keys in CNContact.

7.17.9 showRelativeToRect(positioningRect as NSRectMBS, view as NSViewMBS, edge as Integer)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Shows the picker popover anchored to the specified view.
Notes: positioningRect: The content size of the popover.
positioningView: The view to which the popover should be positioned.
preferredEdge: The edge to which the popover should be anchored to. Can be MinYEdge, MinXEdge, MaxYEdge or MaxXEdge.

7.17.10 Properties

7.17.11 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactPicker object.
(Read and Write property)

7.17.12 Events

7.17.13 DidClose

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: Called when the contact picker‚Äôs popover has closed.

7.17.14 didSelectContact(contact as CNContactMBS)

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: Called after a contact has been selected by the user.
Notes: The contact may only be partial loaded and not have all fields available as the picker may not need/show all fields.

7.17.15 didSelectContactProperty(contactProperty as CNContactPropertyMBS)

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: The event called when a contact property was selected.
Notes: The contact may only be partial loaded and not have all fields available as the picker may not need/show all fields.

7.17.16 WillClose

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: Called when the contact picker‚Äôs popover is about to close.

7.17.17 Constants

Edge Constants

Constant	Value	Description
MaxXEdge	2	
MaxYEdge	3	
MinXEdge	0	
MinYEdge	1	

7.18 class CNContactPickerViewControllerMBS

7.18.1 class CNContactPickerViewControllerMBS

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: A view controller that displays an interface for picking contacts.
Notes: A CNContactPickerViewController allows the user to select one or more contacts (or their properties) from the list of contacts displayed in the contact view controller (CNContactViewController). The picker supports both single selection and multiselection of the contacts. The app using contact picker view does not need access to the user,Äôs contacts and the user will not be prompted for ,Äúgrant permission,Äù access. The app has access only to the user,Äôs final selection.
There are predefined predicates in this class that let you control the user selection of the contact. Changing the predicates only take effect before the view is presented.
Subclass of the NSViewControllerMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3
- CNContactPickerViewController for Xojo
- MBS Xojo Plugins, version 22.3pr2

7.18.2 Methods

7.18.3 available as Boolean

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: Whether this class is available.
Notes: Requires iOS 9.0

7.18.4 Constructor

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: The constructor.

7.18.5 Destructor

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: The destructor.

7.18.6 Dismiss

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: Dismisses the controller.

7.18.7 displayedPropertyKeys as String()

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: The CNContact property keys to display in the contact detail card.
Notes: All the properties of the contact are displayed if this property is not set.

7.18.8 Present

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: Presents the controller.

7.18.9 setdisplayedPropertyKeys(displayedPropertyKeys() as String)

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: The CNContact property keys to display in the contact detail card.
Notes: All the properties of the contact are displayed if this property is not set.

7.18.10 Properties

7.18.11 isBeingPresented as Boolean

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: Whether the controller is presented.
Notes: Returns true if the dialog is visible.
(Read only property)

7.18.12 predicateForEnablingContact as NSPredicateMBS

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: A predicate to determine the contact selectability in the list of contacts.
Notes: You can set a value for this property to determine which contact should become selectable, such as emailAddresses.@count >0 to enable all the contacts that have an email address to become selectable. If no predicate is set for this property, all contacts become selectable. To learn about predicate syntax, see NSPredicateMBS.
(Read and Write property)

7.18.13 predicateForSelectionOfContact as NSPredicateMBS

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: A predicate to control the return of the selected contact.
Notes: This property determines whether a selected contact should be returned (when the predicate evaluates to TRUE), or a default action for the property should be performed (when the predicate evaluates to FALSE). By default the contact picker view controller displays the contact,Äôs detail card when a contact is selected. To learn about predicates, see NSPredicateMBS.
(Read and Write property)

7.18.14 predicateForSelectionOfProperty as NSPredicateMBS

Plugin Version: 22.3, Platform: iOS, Targets: iOS only.

Function: A predicate to control the properties of the selected contact.
Notes: This property determines whether a selected contact should be returned (when the predicate evaluates to TRUE), or a default action for the property should be performed (when the predicate evaluates to FALSE). By default the contact picker view controller returns the first selected property of the contact. This predicate is evaluated on the CNContactProperty property that is being selected, such as (key == 'emailAddresses') AND (value LIKE '*@apple.com') to return email address of the contact if the address contains the string ,Äú@apple.com,Äù.
(Read and Write property)

7.18.15 Events

7.18.16 didCancel

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: In iOS, called when the user taps Cancel.
Notes: The picker is dismissed automatically after a contact or property is picked.

7.18.17 didSelectContact(contact as CNContactMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Called after a contact has been selected by the user.
Notes: This event is called when the user selects a single contact.

7.18.18 didSelectContactProperties(contactProperties() as CNContactPropertyMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Called after contact properties have been selected by the user.
Notes: This event is invoked when the user selects more than one property. Implementing this method configures the picker for multi-selection.

7.18.19 didSelectContactProperty(contactProperty as CNContactPropertyMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Called when a property of the contact has been selected by the user.
Notes: This event ia called when the user selects a single property of the contact.

7.18.20 didSelectContacts(contacts() as CNContactMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Called after contacts have been selected by the user.
Notes: This event is called when the user selects more than one contact. Implementing this method configures the picker for multi-selection.

7.19 class CNContactPropertyMBS

7.19.1 class CNContactPropertyMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactProperty is a convenience class and returns a tuple or quadruple to a contact‚Äôs property.
Notes: A property contains related information for a specific contact. A contact (an instance of CNContact) has properties, such as firstName, phoneNumber, and jobTitle. Each property is represented by an instance of CNContactProperty, which provides a tuple that can contain three or five values, depending on whether the property is a member of an array of labeled values. For example, the phoneNumbers property is a member of an array of labeled values, so the CNContactProperty tuple contains the contact, key, value, identifier, and label. For the givenName property, which is not contained in a labeled array, CNContactProperty returns a tuple that contains the contact, key, and value. The CNContactProperty class is used by the CNContactPicker to return the user's selected property.
Blog Entries

- CNContactPickerViewController for Xojo

7.19.2 Methods

7.19.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.19.4 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.19.5 copy as CNContactPropertyMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the social property object.

7.19.6 Properties

7.19.7 Contact as CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: CNContact property of the selected contact.
Notes: This property is used for properties that may or may not be in labeled arrays.
(Read only property)

7.19.8 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactProperty object.
(Read and Write property)

7.19.9 Identifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The identifier of the labeled value in the array of labeled.
Notes: Identifier is used only for properties in labeled arrays. If the property is not an array of labeled values, the value of the identifier is "".
(Read only property)

7.19.10 Key as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The key of the contact property.
Notes: This property is used for properties that may or may not be in labeled arrays.
(Read only property)

7.19.11 Label as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The label of the labeled value of the property array.
Notes: Labeled property is used only for properties that are in labeled arrays. If the property is not an

array of labeled values, the value of the label is "".
(Read only property)

7.19.12 Value as Variant

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The value of the property.
Notes: This property is used for properties that may or may not be in labeled arrays.
(Read only property)

7.20 class CNContactRelationMBS

7.20.1 class CNContactRelationMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactRelation class defines an immutable value object representing a contact related to another.
Notes: This is a thread-safe class.

7.20.2 Methods

7.20.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.20.4 CNLabelContactRelationAssistant as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Assistant.

7.20.5 CNLabelContactRelationBrother as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Brother

7.20.6 CNLabelContactRelationChild as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Child

7.20.7 CNLabelContactRelationDaughter as String

Plugin Version: 17.4, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Daugther
for macOS 10.13 or newer.

7.20.8 CNLabelContactRelationFather as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Father

7.20.9 CNLabelContactRelationFriend as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Friend

7.20.10 CNLabelContactRelationManager as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Manager

7.20.11 CNLabelContactRelationMother as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Mother

7.20.12 CNLabelContactRelationParent as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Parent

7.20.13 CNLabelContactRelationPartner as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Partner

7.20.14 CNLabelContactRelationSister as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Sister

7.20.15 CNLabelContactRelationSon as String

Plugin Version: 17.4, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Son
for macOS 10.13 or newer.

7.20.16 CNLabelContactRelationSpouse as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the relation values.
Notes: Spouse

7.20.17 Constructor(name as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initialize a class instance with the name of the related contact.

7.20.18 contactRelationWithName(name as string) as CNContactRelationMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new contact relation with a name.

7.20.19 copy as CNContactRelationMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the contact relation object.

7.20.20 Properties

7.20.21 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactRelation object.
(Read and Write property)

7.20.22 Name as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the related contact.
Notes: (Read only property)

7.21 class CNContactStoreMBS

7.21.1 class CNContactStoreMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactStore class is a thread-safe class that can fetch and save contacts, groups, and containers.
Notes: The CNContactStore class provides ways to execute fetch and save requests. There are a few recommended ways you can implement these requests in your app to load contacts:

- Fetch only the contact properties that will be used.
- When fetching all contacts and caching the results, first fetch all contacts identifiers, then fetch batches of detailed contacts by identifiers as required.
- To aggregate several contacts fetches, first collect a set of unique identifiers from the fetches. Then fetch batches of detailed contacts by those unique identifiers.
- If you cache the fetched contacts, groups, or containers, you need to refetch these objects (and release the old cached objects) when CNContactStoreDidChangeNotification is posted.

Because CNContactStore fetch methods perform I/O, it,Äôs recommended that you avoid using the main thread to execute fetches.

Your app must be code signed to see contacts.

Warning: Do not use in macOS 10.11 or earlier due to bugs, which have been fixed in macOS 10.12.
Blog Entries

- News from the MBS Xojo Plugins Version 22.0
- Contacts history for Xojo
- MBS Xojo Plugins, version 20.5pr1
- MBS Xojo Plugins, version 19.0pr8
- MBS Xojo Plugins, version 18.4pr7
- MBS Xojo Plugins, version 18.4pr5
- MBS Xojo Plugins, version 18.1pr1

7.21.2 Methods

7.21.3 allContacts(byref error as NSErrorMBS) as CNContactMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Searches all contacts.
Notes: Same as ContactsWithFetchRequest function with key list from CNContactStoreMBS.AllFetchKeys function.

7.21.4 AllFetchKeys as String()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns list of all keys used by the plugin to query all properties for contacts.

7.21.5 authorizationStatusForEntityType(entityType as Integer = 0) as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the current authorization status to access the contact data.
Notes: entityType: Set to CNEntityType, e.g. CNEntityTypeContacts.

Returns the current authorization status to access the contact data.

Based on the authorization status, your application might display or hide its UI elements that access any Contacts API. This method is thread-safe and will not block your application. To see different authorization status, see CNAuthorizationStatus.

7.21.6 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.21.7 CNContactStoreDidChangeNotification as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The notification posted when changes occur in another CNContactStore.

7.21.8 CNErrorDomain as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The error domain for NSErrorMBS.

7.21.9 CNErrorUserInfoAffectedRecordIdentifiersKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A key for the NSError userInfo dictionary.
Notes: When available an array of one or more NSString objects for which the error code applies.

7.21.10 CNErrorUserInfoAffectedRecordsKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A key for the NSError userInfo dictionary.
Notes: When available an array of one or more CNContact, CNGroup or CNContainer objects for which the error code applies.

7.21.11 CNErrorUserInfoKeyPathsKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A key for the NSError userInfo dictionary.
Notes: An array of key paths associated with a given error. For validation errors this will contain key paths to specific object properties.

7.21.12 CNErrorUserInfoValidationErrorsKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A key for the NSError userInfo dictionary.
Notes: An array of NSErrors for CNErrorCodeValidationMultipleErrors.

7.21.13 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.21.14 ContactsWithFetchRequest(fetchRequest as CNContactFetchRequestMBS, byref error as NSErrorMBS) as CNContactMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns all contacts matching a contact fetch request.
Notes: fetchRequest: The contact fetch request that specifies the search criteria.
error: Error information, if an error occurred.

This method waits until the enumeration is finished. If there are no results, the method returns an empty array.
This can be used to fetch all contacts without keeping all of them at once in memory because this is expensive.

Your app must be code signed to see contacts.

7.21.15 containersMatchingPredicate(predicate as NSPredicateMBS, byref error as NSErrorMBS) as CNContainerMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all containers matching the specified predicate.
Notes: predicate: The predicate to use to fetch matching containers. Set this property to nil to match all containers.
error: Error information, if an error occurred.

Returns an array of CNContainer objects that match the predicate.

A container holds a collection of contacts, a contact (each contact can be in only one container). CardDAV accounts usually have only one container of contacts. Exchange accounts may have multiple containers, where each container represents an Exchange folder.

This method returns an empty array when no matching container is found. In case of an error this method returns nil. You should use only the predicates defined CNContainer class. Compound predicates are not supported.

7.21.16 defaultContainerIdentifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the identifier of the default container.
Notes: This identifier can be used to fetch a default container. A default container is where the user wants new contacts to be added implicitly.

7.21.17 Destructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

7.21.18 enumerateContactsWithFetchRequest(fetchRequest as CNContactFetchRequestMBS, byref error as NSErrorMBS, tag as Variant = nil) as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a Boolean value that indicates whether the enumeration of all contacts matching a contact fetch request executed successfully.
Notes: fetchRequest: The contact fetch request that specifies the search criteria.
error: Error information, if an error occurred.

Calls enumerateContactsWithFetchRequest event for each contact found.

Returns true if enumeration of all contacts matching a contact fetch request executes successfully; otherwise, false.

This method waits until the enumeration is finished. If there are no results, the event is not called and the method returns true.

This can be used to fetch all contacts without keeping all of them at once in memory because this is expensive.
See also:

- 7.21.37 enumerateContactsWithFetchRequest(contact as CNContactMBS, byref stop as boolean, tag as Variant) 493

7.21.19 enumeratorForChangeHistoryFetchRequest(request as CNChangeHistoryFetchRequestMBS, byref error as NSErrorMBS) as CNFetchResultMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: Enumerate a change history fetch request.
Notes: Executes the given fetch request and returns an enumerator for the results.
This may prevent all events from being loaded into memory at once.

An exception may be thrown if an error occurs during enumeration.

request: A description of the events to fetch.
error: If the fetch fails, contains an NSErrorMBS object with more information.

Returns an enumerator of the events matching the result, or nil if there was an error.

7.21.20 enumeratorForContactFetchRequest(request as CNContactFetchRequestMBS, byref error as NSErrorMBS) as CNFetchResultMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: Enumerate a contact fetch request.
Notes: Executes the given fetch request and returns an enumerator for the results.
This may prevent all records from being loaded into memory at once.

An exception may be thrown if an error occurs during enumeration.

request: A description of the records to fetch.
error: If the fetch fails, contains an NSErrorMBS object with more information.

An enumerator of the records matching the result, or nil if there was an error.

7.21.21 executeSaveRequest(saveRequest as CNSaveRequestMBS, byref Error as NSErrorMBS) as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Executes a save request and returns success or failure.
Notes: saveRequest: The save request to execute.

error: Error information, if an error occurred.

Returns true if the save request executes successfully; otherwise, false.

It is recommended that you do not access objects in the save request from other threads when it is in the process of being executed, because it may modify the contacts in the process. A save request only applies the changes to the objects. If there are overlapping changes with multiple or concurrent CNSaveRequest then the last saved change wins.

Warning: Do not use in macOS 10.11 or earlier due to bugs, which have been fixed in macOS 10.12.

7.21.22 groupsByName(name as String, byref error as NSErrorMBS) as CNGroupMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Looks for groups with matching names.
Notes: Returns empty array if nothing found.
Comparison is localized case insensitive.

7.21.23 groupsForContact(contact as CNContactMBS, byref error as NSErrorMBS) as CNGroupMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Searches all groups for a given contact.

7.21.24 groupsMatchingPredicate(predicate as NSPredicateMBS, byref error as NSErrorMBS) as CNGroupMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all groups matching the specified predicate.
Notes: predicate: The predicate to use to fetch the matching groups. Set predicate to nil to match all groups.
error: Error information, if an error occurred.

Returns an array of CNGroup objects that match the predicate.

This method returns an empty array when no matching groups are found. If an error occurs, this method returns nil. You should use only the predicates defined in CNGroup class predicates. Compound predicates are not supported. Contacts may be members of one or more groups, depending upon the account they come from.

7.21.25 requestAccessForEntityType(entityType as Integer = 0, tag as Variant = nil)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Requests access to the user's contacts.
Notes: entityType: Set to CNEntityTypeContacts.

Users are able to grant or deny access to contact data on a per-application basis. Request access to contact data by calling requestAccessForEntityType method. This will not block your application while the user is being asked for permission. The user will only be prompted the first time access is requested; any subsequent CNContactStore calls will use the existing permissions. The requestAccessForEntityType event is later called. This method is optional when CNContactStore is used in the background thread. If this method is not used, CNContactStore may block your application while the user is asked for access permission.

Your app must be code signed to see contacts.
See also:

- 7.21.38 requestAccessForEntityType(granted as boolean, error as NSErrorMBS, tag as Variant) 493

7.21.26 unifiedContactsMatchingPredicate(predicate as NSPredicateMBS, byref error as NSErrorMBS) as CNContactMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all unified contacts matching the specified predicate.
Notes: predicate: The predicate to match against.
error: Error information, if an error occurred.

Returns an array of CNContact objects matching the predicate.

If no matches are found, this method returns an empty array (or nil in case of error). Use only the predicates from the CNContact class predicates. Compound predicates are not supported by this method. Due to unification, the returned contacts may have different identifiers than you specify. To fetch all contacts, use enumerateContactsWithFetchRequest.

Same as other unifiedContactsMatchingPredicate function with key list from CNContactStoreMBS.AllFetchKeys

function.
See also:

- 7.21.27 unifiedContactsMatchingPredicate(predicate as NSPredicateMBS, keysToFetch() as CNKeyDescriptorMBS, byref error as NSErrorMBS) as CNContactMBS() 490

7.21.27 unifiedContactsMatchingPredicate(predicate as NSPredicateMBS, keysToFetch() as CNKeyDescriptorMBS, byref error as NSErrorMBS) as CNContactMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches all unified contacts matching the specified predicate.
Notes: predicate: The predicate to match against.
keys: The properties to fetch in the returned CNContact objects. You should fetch only the properties that you plan to use. Note that you can combine contact keys and contact key descriptors.
error: Error information, if an error occurred.

Returns an array of CNContact objects matching the predicate.

If no matches are found, this method returns an empty array (or nil in case of error). Use only the predicates from the CNContact class predicates. Compound predicates are not supported by this method. Due to unification, the returned contacts may have different identifiers than you specify. To fetch all contacts, use enumerateContactsWithFetchRequest.
See also:

- 7.21.26 unifiedContactsMatchingPredicate(predicate as NSPredicateMBS, byref error as NSErrorMBS) as CNContactMBS() 489

7.21.28 unifiedContactWithIdentifier(identifier as string, byref error as NSErrorMBS) as CNContactMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches a unified contact for the specified contact identifier.
Notes: identifier: The identifier of the contact to fetch.
error: Error information, if an error occurred.

Returns an unified contact matching or linked to the identifier.

Due to unification, the returned contact may have a different identifier, than you specify. To fetch a batch of contacts by identifiers use predicateForContactsWithIdentifiers with unifiedContactsMatchingPredicate. It is recommended to fetch only the properties that will be used. You can combine contact keys and contact

key descriptors together.

Same as other unifiedContactWithIdentifier function with key list from CNContactStoreMBS.AllFetchKeys function.
See also:

- 7.21.29 unifiedContactWithIdentifier(identifier as string, keys() as CNKeyDescriptorMBS, byref error as NSErrorMBS) as CNContactMBS 491

7.21.29 unifiedContactWithIdentifier(identifier as string, keys() as CNKeyDescriptorMBS, byref error as NSErrorMBS) as CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Fetches a unified contact for the specified contact identifier.
Notes: identifier: The identifier of the contact to fetch.
keys: The properties to fetch in the returned CNContact object.
error: Error information, if an error occurred.

Returns an unified contact matching or linked to the identifier.

Due to unification, the returned contact may have a different identifier, than you specify. To fetch a batch of contacts by identifiers use predicateForContactsWithIdentifiers with unifiedContactsMatchingPredicate. It is recommended to fetch only the properties that will be used. You can combine contact keys and contact key descriptors together.
See also:

- 7.21.28 unifiedContactWithIdentifier(identifier as string, byref error as NSErrorMBS) as CNContactMBS 490

7.21.30 unifiedMeContact(byref error as NSErrorMBS) as CNContactMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: Queries an unified contact for the user.
Example:

```
dim ContactStore as CNContactStoreMBS // your store
Dim error As NSErrorMBS
Dim contact As CNContactMBS = ContactStore.unifiedMeContact(Error)

If contact <>nil then
MsgBox contact.familyName
Else
MsgBox "No contact?"
```

```
End If
```

Notes: Same as unifiedMeContactWithKeysToFetch function with key list from CNContactStoreMBS.AllFetchKeys function.

7.21.31 unifiedMeContactWithKeysToFetch(keys() as CNKeyDescriptorMBS, byref error as NSErrorMBS) as CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Queries an unified contact for the user.

7.21.32 Properties

7.21.33 currentHistoryToken as MemoryBlock

Plugin Version: 22.0, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieve the current history token.
Notes: If you are fetching contacts or change history events, you should use the token on the CNFetchResultMBS class instead.
(Read only property)

7.21.34 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactStore object.
(Read and Write property)

7.21.35 Events

7.21.36 DidChange

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: The event posted when changes occur in another CNContactStore.

7.21.37 enumerateContactsWithFetchRequest(contact as CNContactMBS, byref stop as boolean, tag as Variant)

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: The event called by enumerateContactsWithFetchRequest for each new contact found.
See also:

- 7.21.18 enumerateContactsWithFetchRequest(fetchRequest as CNContactFetchRequestMBS, byref error as NSErrorMBS, tag as Variant = nil) as Boolean 486

7.21.38 requestAccessForEntityType(granted as boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 16.3, Platform: macOS, Targets: .

Function: The request for access completed.
Notes: Set granted to true if the user allows access and error is nil.
See also:

- 7.21.25 requestAccessForEntityType(entityType as Integer = 0, tag as Variant = nil) 489

7.21.39 Constants

Constants

Constant	Value	Description
CNAuthorizationStatusAuthorized	3	One of the authorization status values the user can grant for an app to the specified entity type. The application is authorized to access contact data.
CNAuthorizationStatusDenied	2	One of the authorization status values the user can grant for an app to the specified entity type. The user explicitly denied access to contact data for the application.
CNAuthorizationStatusNotDetermined	0	One of the authorization status values the user can grant for an app to the specified entity type. The user has not yet made a choice regarding whether the applicatio access contact data.
CNAuthorizationStatusRestricted	1	One of the authorization status values the user can grant for an app to the specified entity type. The application is not authorized to access contact data. The user c change this application,Äôs status, possibly due to active restrictions s parental controls being in place.

Contact Types

Constant	Value	Description
CNEntityTypeContacts	0	Contacts

Error Codes

Constant	Value	Description
CNErrorCodeAuthorizationDenied	100	Authentication denied error.
CNErrorCodeCommunicationError	1	Communication error.
CNErrorCodeContainmentCycle	202	Code containment cycle error.
CNErrorCodeContainmentScope	203	Code containment scope error.
CNErrorCodeDataAccessError	2	Data access error.
CNErrorCodeInsertedRecordAlreadyExists	201	Record already exists.
CNErrorCodeParentRecordDoesNotExist	204	The contact does not exist error.
CNErrorCodePolicyViolation	500	Policy validation error.
CNErrorCodePredicateInvalid	400	Invalid predicate error.
CNErrorCodeRecordDoesNotExist	200	Record not found.
CNErrorCodeRecordIdentifierInvalid	205	Record identifier not valid.
CNErrorCodeValidationConfigurationError	302	Configuration validation error.
CNErrorCodeValidationMultipleErrors	300	Multiple validation error.
CNErrorCodeValidationTypeMismatch	301	Type mismatch validation error.

7.22 class CNContactsUserDefaultsMBS

7.22.1 class CNContactsUserDefaultsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactsUserDefaults class defines properties used to access the user defaults for a contact.

7.22.2 Methods

7.22.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.22.4 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.22.5 sharedDefaults as CNContactsUserDefaultsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The singleton instance of the CNContactsUserDefaults class.

7.22.6 Properties

7.22.7 countryCode as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An ISO country code.
Notes: ISO is the default country code for phone numbers. This is determined by the device,Äôs SIM card or the operating system,Äôs configured language.

(Read only property)

7.22.8 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContactsUserDefaults object.
(Read and Write property)

7.22.9 sortOrder as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Default sorting order by name.
Notes: Sort order is determined by the operating system,Äôs configured language or overridden by the user.
(Read only property)

7.23 class CNContactVCardSerializationMBS

7.23.1 class CNContactVCardSerializationMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactVCardSerialization supports vCard representation for the given set of contacts.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.23.2 Methods

7.23.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.23.4 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

7.23.5 contactsWithData(Data as MemoryBlock, byref error as NSErrorMBS) as CNContactMBS()

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the contacts from the vCard data.

7.23.6 dataWithContacts(Contacts() as CNContactMBS, byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the vCard representation of the specified contacts.

7.23.7 descriptorForRequiredKeys as CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Use to fetch all contact keys required to create vCard data from a contact.
Notes: A key descriptor to be used in the keysToFetch array when fetching the contacts.

7.24 class CNContactViewControllerMBS

7.24.1 class CNContactViewControllerMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContactViewController class implements the view to display a contact.
Notes: CNContactViewController can display a new contact, unknown contact, or existing contact.
Subclass of the NSViewControllerMBS class.
Blog Entries

- MBS Xojo Plugins, version 17.5pr3

7.24.2 Methods

7.24.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.24.4 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.24.5 descriptorForRequiredKeys as CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Descriptor for all keys that must be fetched on a contact before setting it on the view controller.
Notes: Pass this descriptor to the keysToFetch of the CNContactFetchRequest if you want to display the contact in a CNContactViewController.

7.24.6 Properties

7.24.7 Contact as CNContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A contact to display.
Notes: When contact is nil, displays an empty selection state.
(Read and Write property)

7.25 class CNContainerMBS

7.25.1 class CNContainerMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNContainer class is a thread-safe class that defines an immutable object that represents a container.
Notes: A container has a collection of contacts. A contact can be in only one container. CardDAV accounts usually have only one container whereas Exchange accounts may have multiple containers, where each container represents an Exchange folder
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.25.2 Methods

7.25.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.25.4 CNContainerIdentifierKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined keys representing container properties that can be used with key value coding on CNContainer objects.
Notes: Identifier key.
This key represents the container identifier property for KVC/KVO usage. This property is always fetched.

7.25.5 CNContainerNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined keys representing container properties that can be used with key value coding on CNContainer objects.
Notes: Name key.
This key represents the container identifier property for KVC/KVO usage. This property is always fetched.

7.25.6 CNContainerTypeKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined keys representing container properties that can be used with key value coding on CNContainer objects.
Notes: Type key.
This key represents the container identifier property for KVC/KVO usage. This property is always fetched.

7.25.7 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private container.

7.25.8 copy as CNContainerMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the container object.

7.25.9 predicateForContainerOfContactWithIdentifier(contactIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the container of the specified contact.
Notes: If the identifier is for a unified contact then this method returns an empty array. To fetch the containers of a unified contact, first fetch the linked contacts and then fetch the container of each linked contact.

7.25.10 predicateForContainerOfGroupWithIdentifier(groupIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the container of the specified group.

7.25.11 predicateForContainersWithIdentifiers(Identifiers() as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find the containers with the specified identifiers.

7.25.12 Properties

7.25.13 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNContainer object.
(Read and Write property)

7.25.14 Identifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The unique identifier for a contacts container on the device.
Notes: It is recommended that you use the identifier when re-fetching the container. The identifier can be persisted between app launches.
(Read only property)

7.25.15 Name as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the container.
Notes: (Read only property)

7.25.16 Type as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The type of the container.
Notes: can be CNContainerTypeUnassigned, CNContainerTypeLocal, CNContainerTypeExchange or CNContainerTypeCardDAV.

(Read only property)

7.25.17 Constants

Container Types

Constant	Value	Description
CNContainerTypeCardDAV	3	A container for contacts stored in an CardDAV server, such as iCloud.
CNContainerTypeExchange	2	A container for contacts stored in an Exchange folder from an Exchange server.
CNContainerTypeLocal	1	A container for contacts only stored locally on the device. There is only one local container for a device.
CNContainerTypeUnassigned	0	

7.26 class CNFetchResultMBS

7.26.1 class CNFetchResultMBS

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The class for a fetch result with a time stamp.
Notes: Allows you to query contacts and changes with checking again later with new time.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.0
- Contacts history for Xojo

7.26.2 Methods

7.26.3 ChangeHistoryEvents as CNChangeHistoryEventMBS()

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The history events.

7.26.4 Constructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

7.26.5 Contacts as CNContactMBS()

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The contacts found.

7.26.6 Destructor

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The destructor.

7.26.7 Properties

7.26.8 currentHistoryToken as MemoryBlock

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The current history token.
Notes: Retrieve the current history token. If you are fetching contacts or change history events, you should use the token on the CNFetchResultMBS class instead.
(Read only property)

7.27 class CNGroupMBS

7.27.1 class CNGroupMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNGroup is a thread-safe class that defines an immutable object that represents a group.
Notes: Contacts may be members of one or more groups, depending upon their accounts.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 17.2pr5

7.27.2 Methods

7.27.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.27.4 CNGroupIdentifierKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined keys representing group properties that can be used with key value coding on CNGroup objects.
Notes: Group identifier.
This key takes a string value.

7.27.5 CNGroupNameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined keys representing group properties that can be used with key value coding on CNGroup objects.
Notes: Group name.
This key takes a string value.

7.27.6 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

7.27.7 copy as CNGroupMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the group object.

7.27.8 mutableCopy as CNMutableGroupMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a mutable copy of this group.

7.27.9 predicateForGroupsInContainerWithIdentifier(groupIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find groups in the specified container.
Notes: containerIdentifier: The container identifier to be matched.

Returns a predicate that can be used to fetch groups from CNContactStore.

7.27.10 predicateForGroupsWithIdentifiers(Identifiers() as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a predicate to find groups with the specified identifiers.
Notes: identifiers: The group identifiers to be matched.

Returns a predicate that can be used to fetch groups from CNContactStore.

7.27.11 predicateForSubgroupsInGroupWithIdentifier(contactIdentifier as String) as NSPredicateMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a predicate to find all subgroups for a group.

7.27.12 valueForKey(key as String) as Variant

Plugin Version: 17.2, Platform: macOS, Targets: Desktop & iOS.

Function: Queries value for a given key.

7.27.13 Properties

7.27.14 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNGroup object.
(Read and Write property)

7.27.15 Identifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The unique identifier for a group on the device.
Notes: It is recommended that you use the identifier when re-fetching the group. The identifier can be persisted between app launches.
(Read only property)

7.27.16 Name as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the group.
Notes: (Read only property)

7.28 class CNInstantMessageAddressMBS

7.28.1 class CNInstantMessageAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNInstantMessageAddress class is a thread-safe class that defines an immutable value object representing an instant message address.
Notes: Use the methods and properties of this class to identify instant messaging address. Some instant message services, such as Facebook and Skype are predefined in this class. You can also specify your own instant message service using Constructor.

7.28.2 Methods

7.28.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.28.4 CNInstantMessageAddressServiceKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Instant message address service key.
Notes: This key takes a string value.

7.28.5 CNInstantMessageAddressUsernameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Instant message address user name key.
Notes: This key takes a string value.

7.28.6 CNInstantMessageServiceAIM as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for AIM.

This key takes a string value.

7.28.7 CNInstantMessageServiceFacebook as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Facebook.
This key takes a string value.

7.28.8 CNInstantMessageServiceGaduGadu as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Gadu Gadu.
This key takes a string value.

7.28.9 CNInstantMessageServiceGoogleTalk as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Google Talk.
This key takes a string value.

7.28.10 CNInstantMessageServiceICQ as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for ICQ.
This key takes a string value.

7.28.11 CNInstantMessageServiceJabber as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Jabber.

This key takes a string value.

7.28.12 CNInstantMessageServiceMSN as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for MSN.
This key takes a string value.

7.28.13 CNInstantMessageServiceQQ as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for QQ.
This key takes a string value.

7.28.14 CNInstantMessageServiceSkype as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Skype.
This key takes a string value.

7.28.15 CNInstantMessageServiceYahoo as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the service types.
Notes: Instant message service for Yahoo.
This key takes a string value.

7.28.16 Constructor(username as String, Service as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a CNInstantMessageAddress object initialized with the specified user name and service.
Notes: username: The user name with which to initialize the CNInstantMessageAddress object.

service: The service with which to Initialize the CNInstantMessageAddress object.

Returns the initialized CNInstantMessageAddress object with the specified user name and service. User name and service are required to initialize CNInstantMessageAddress object.

7.28.17 copy as CNInstantMessageAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the instant message address object.

7.28.18 localizedStringForKey(key as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a string containing the localized property name.

7.28.19 localizedStringForService(key as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a string containing the localized name of the specified service.

7.28.20 Properties

7.28.21 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNInstantMessageAddress object.
(Read and Write property)

7.28.22 service as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Instant message address service.
Notes: (Read only property)

7.28.23 username as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The user name for instant message service address.
Notes: (Read only property)

7.29 class CNKeyDescriptorMBS

7.29.1 class CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The class for key descriptors.
Blog Entries

- Contacts history for Xojo

7.29.2 Methods

7.29.3 Constructor(Key as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a key descriptor from a string value.

7.29.4 copy as CNKeyDescriptorMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the key descriptor object.

7.29.5 Operator_Convert as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Converts key to string value.
Notes: Same as using StringValue property, but automatic.
See also:

- 7.29.6 Operator_Convert(Key as String) 515

7.29.6 Operator_Convert(Key as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a key descriptor from a string value.
Example:

```
dim k as CNKeyDescriptorMBS = CNContactMBS.CNContactJobTitleKey
```

Notes: Same as constructor, but automatic.
See also:

- 7.29.5 Operator_Convert as String 515

7.29.7 Properties

7.29.8 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNKeyDescriptor object.
(Read and Write property)

7.29.9 StringValue as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries string value of the key descriptor.
Notes: (Read only property)

7.30 class CNLabeledValueMBS

7.30.1 class CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The class for labeled values.
Notes: The CNLabeledValue class is a thread-safe class that defines an immutable value object that combines a contact property value with a label. For example, a contact phone number could have a label of Home, Work, iPhone, etc.
Blog Entries

- MBS Xojo Plugins, version 17.1pr4

7.30.2 Methods

7.30.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.30.4 CNLabelDateAnniversary as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: This constant is a predefined label that can be used in a CNLabeledValue object having an NSDateComponents value.

7.30.5 CNLabelEmailiCloud as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: This constant is a predefined label that can be used in a CNLabeledValue object having an email address string value.
Notes: Email.

This label takes a string value.

7.30.6 CNLabelHome as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined labels that can be used in a CNLabeledValue object having any value.
Notes: Home label.
This label takes a string value.

7.30.7 CNLabelOther as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined labels that can be used in a CNLabeledValue object having any value.
Notes: Other label.
This label takes a string value.

7.30.8 CNLabelURLAddressHomePage as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: This constant is a predefined label that can be used in a CNLabeledValue object having a URL address string value.
Notes: Identifier for the URL property.
This label takes a string value.

7.30.9 CNLabelWork as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the predefined labels that can be used in a CNLabeledValue object having any value.
Notes: Work label.
This label takes a string value.

7.30.10 Constructor(label as string, value as Variant)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new labeled value identifier initialized with the specified label and value.
Notes: label: A string value for the label of the labeled ,Ä¶value object, or nil if the value doesn,Äôt have a label.
value: A value for the labeled value object. For valid values, see CNContact properties that are arrays of

labeled value objects.

Returns a new labeled value object initialized with the specified identifier.

7.30.11 copy as CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the labeled value object.

7.30.12 labeledValueBySettingLabel(label as string) as CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a labeled value object with an existing value and identifier.
Notes: label: The label of the copied labeled value object, or nil if the contact property value doesn,Äôt have a label.

Returns a labeled value object with an existing value and identifier.
See also:

- 7.30.13 labeledValueBySettingLabel(label as string, value as Variant) as CNLabeledValueMBS 519

7.30.13 labeledValueBySettingLabel(label as string, value as Variant) as CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a labeled value object with the specified label and value with the existing identifier.
Notes: label: The label of the copied labeled value object, or "" if the contact property value doesn,Äôt have a label.
value: The copied labeled value object. For valid values, see CNContact properties that are arrays of labeled value objects.

Returns a labeled value object with the existing identifier.
See also:

- 7.30.12 labeledValueBySettingLabel(label as string) as CNLabeledValueMBS 519

7.30.14 labeledValueBySettingValue(value as Variant) as CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new value for an existing label and identifier.
Notes: value: A new value for the copied labeled value object. For valid values, see CNContact properties that are arrays of labeled value objects.

Returns the CNLabeledValue object with an existing label and identifier.

7.30.15 labeledValueWithLabel(label as string, value as Variant) as CNLabeledValueMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new labeled value identifier object with the specified label and value.
Notes: label: A string value for the label of the labeled value object, or nil if the value doesn,Äôt have a label.
value: A value for the labeled value object. For valid values, see CNContact properties that are arrays of labeled value objects.

Returns a new CNLabeledValue object with a new identifier.

7.30.16 localizedStringForLabel(label as string) as string

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a localized string for the specified label.
Notes: label: The label to be localized.

Returns a localized string for the label.

All predefined label constants are localized and this method returns their localized strings. A custom label will be returned as is, so this method can be used to convert all labels for display.

7.30.17 Properties

7.30.18 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNLabeledValue object.
(Read and Write property)

7.30.19 Identifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A unique identifier for the labeled value object.
Notes: It is recommended that you use the identifier when searching for a previously known labeled value object in a re-fetched contact. The identifier can be persisted between the app launches.
(Read only property)

7.30.20 Label as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The label for a contact property value.
Notes: A contact property can have a label, such as Home, Work, iPhone, etc. For some predefined label constants, seeCNPhoneNumber, and CNContactRelation. Custom labels can also be used. Labels are not used for CNSocialProfile and CNInstantMessageAddress properties.
(Read only property)

7.30.21 Value as Variant

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A contact property value.
Notes: A contact property value, such as CNPhoneNumberMBS for a phone number, String for an email address, and so on. For valid values, see CNContact properties that are arrays of labeled value objects.
(Read only property)

7.31 class CNMutableContactMBS

7.31.1 class CNMutableContactMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNMutableContact class represents a mutable value object for the contact properties, such as the first name and the phone number of a contact.
Notes: When CNMutableContact object is a mutable copy of a CNContact object, if you access a CNMutableContact property value that was not fetched for the CNContact object, it throws an CNContactPropertyNotFetchedExceptionName exception. When needed, you can remove contact properties by setting string and array properties to empty, and all other properties to nil.
Available in OS X v10.11 and later.
Subclass of the CNContactMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.5pr1

7.31.2 Methods

7.31.3 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.31.4 setContactRelations(contactRelations() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled contact relations for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNContactRelationMBS value.

7.31.5 setDates(dates() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array containing labeled Gregorian dates.
Notes: This property is an array of CNLabeledValue objects, each of which has an String label and NSDateComponentsMBS value. You can use this property to store Gregorian dates such as anniversaries. Day and month are required and year is optional. Calendar is nil or Gregorian. All other date components are invalid.

7.31.6 setEmailAddresses(emailAddresses() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled email addresses for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a String value.

7.31.7 setInstantMessageAddresses(instantMessageAddresses() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled IM addresses for the contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNInstantMessageAddressMBS value.

7.31.8 setPhoneNumbers(phoneNumbers() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled phone numbers for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNPhoneNumberMBS value.

7.31.9 setPostalAddresses(postalAddresses() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled postal addresses for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNPostalAddressMBS value.

7.31.10 setSocialProfiles(socialProfiles() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled social profiles for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a CNSocialProfileMBS value.

7.31.11 setURLAddresses(urlAddresses() as CNLabeledValueMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the array of labeled URL addresses for a contact.
Notes: This property is an array of CNLabeledValue objects, each of which has a label and a string value that contains the URL.

7.31.12 Properties

7.31.13 birthday as NSDateComponentsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A date component for the Gregorian birthday of the contact.
Notes: A Gregorian birthday can be displayed using this property, whose values are the relevant properties of an NSDateComponents object. Day and month are required for this property, and year is optional. Calendar can be nil or Gregorian. All other date components are invalid.
(Read and Write property)

7.31.14 contactType as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: An enum identifying the contact type.
Notes: Can be CNContactMBS.CNContactTypeOrganization or CNContactMBS.CNContactTypePerson.
(Read and Write property)

7.31.15 departmentName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the department associated with the contact.
Notes: (Read and Write property)

7.31.16 familyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The family name of the contact.
Example:

```
dim m as new CNMutableContactMBS
m.familyName = "Miller"
msgbox "name: "+m.familyName
```

Notes: The family name is often known as the last name of the contact.
(Read and Write property)

7.31.17 givenName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The given name of the contact.
Notes: The given name is often known as the first name of the contact.
(Read and Write property)

7.31.18 imageData as MemoryBlock

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The profile picture of a contact.
Notes: It is recommended that you fetch this property only when you need to access its value, such as when you need to display the contact,Äôs profile picture.
(Read and Write property)

7.31.19 jobTitle as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The contact,Äôs job title.
Notes: (Read and Write property)

7.31.20 middleName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The middle name of the contact.
Notes: (Read and Write property)

7.31.21 namePrefix as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name prefix of the contact.
Notes: (Read and Write property)

7.31.22 nameSuffix as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name suffix of the contact.
Notes: (Read and Write property)

7.31.23 nickname as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The nickname of the contact.
Notes: (Read and Write property)

7.31.24 nonGregorianBirthday as NSDateComponentsMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A date component for the non-Gregorian birthday of the contact.
Notes: A non-Gregorian birthday such as Lunisolar birthdays can be displayed using this property, whose values are the relevant properties of an NSDateComponents object. Day and month are required; year and leapMonth are optional. The calendar property is also required and must be non-Gregorian. Some supported calendars are Buddhist, Chinese, and Islamic. All other date components are invalid.
(Read and Write property)

7.31.25 note as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: A string containing notes for the contact.
Notes: (Read and Write property)

7.31.26 organizationName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the organization associated with the contact.
Notes: (Read and Write property)

7.31.27 phoneticFamilyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic family name of the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs family (or last) name.
(Read and Write property)

7.31.28 phoneticGivenName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic given name of the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs given (or first) name.
(Read and Write property)

7.31.29 phoneticMiddleName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic middle name of the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs middle name.
(Read and Write property)

7.31.30 phoneticOrganizationName as String

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The phonetic name of the organization associated with the contact.
Notes: This property contains a string that specifies the pronunciation of the contact,Äôs organization name.
Requires macOS 10.12 or newer.

(Read and Write property)

7.31.31 previousFamilyName as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The previous family name of the contact.
Notes: The previous family name is often known as the maiden name of the contact.
(Read and Write property)

7.31.32 valueForKey(key as String) as Variant

Plugin Version: 17.2, Platform: macOS, Targets: Desktop & iOS.

Function: Queries or sets value for a given key.
Notes: (Read and Write computed property)

7.32 class CNMutableGroupMBS

7.32.1 class CNMutableGroupMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNMutableGroup class defines a mutable value object representing a group for a contact.
Notes: Contacts may be members of one or more groups, depending upon the accounts they come from.
Subclass of the CNGroupMBS class.

7.32.2 Methods

7.32.3 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.32.4 Properties

7.32.5 Name as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Name of the group.
Example:

```
dim m as new CNMutableGroupMBS
m.Name = "Hello"
```

Notes: (Read and Write property)

7.32.6 valueForKey(key as String) as Variant

Plugin Version: 17.2, Platform: macOS, Targets: Desktop & iOS.

Function: Queries or sets value for a given key.
Notes: (Read and Write computed property)

7.33 class CNMutablePostalAddressMBS

7.33.1 class CNMutablePostalAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNMutablePostalAddress class defines a mutable value object representing the postal address for a contact.
Notes: To remove properties when saving a mutable postal address, set string properties to empty values.
Subclass of the CNPostalAddressMBS class.
Xojo Developer Magazine

- 18.5, page 83: Maps (Part 6), A deep dive into Annotations by Markus Winter

7.33.2 Methods

7.33.3 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.33.4 Properties

7.33.5 City as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The city name.
Example:

```
dim m as new CNMutablePostalAddressMBS
m.City = "New York"
MsgBox m.city
```

Notes: (Read and Write property)

7.33.6 Country as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The country name.
Notes: (Read and Write property)

7.33.7 ISOCountryCode as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The ISO country code.
Notes: (Read and Write property)

7.33.8 PostalCode as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The postal code.
Notes: (Read and Write property)

7.33.9 State as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The state name.
Notes: (Read and Write property)

7.33.10 Street as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The street name.
Notes: A multiline address is delimited with carriage returns (that is, ,Äú\n,Äù).
(Read and Write property)

7.34 class CNPhoneNumberMBS

7.34.1 class CNPhoneNumberMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNPhoneNumber class defines an immutable value object representing a phone number for a contact.
Notes: It is a thread-safe class.
Blog Entries

- MBS Xojo Plugins, version 21.1pr1

7.34.2 Methods

7.34.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.34.4 CNLabelPhoneNumberAppleWatch as String

Plugin Version: 21.1, Platform: macOS, Targets: Desktop & iOS.

Function: Phone number for Apple Watch.
Notes: This label takes a string value.
Available in macOS 11.1 or newer.

7.34.5 CNLabelPhoneNumberHomeFax as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Home fax number.
Notes: This label takes a string value.

7.34.6 CNLabelPhoneNumberiPhone as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: iPhone number.
Notes: This label takes a string value.

7.34.7 CNLabelPhoneNumberMain as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Main phone number.
Notes: This label takes a string value.

7.34.8 CNLabelPhoneNumberMobile as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Mobile phone number.
Notes: This label takes a string value.

7.34.9 CNLabelPhoneNumberOtherFax as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Other fax number.
Notes: This label takes a string value.

7.34.10 CNLabelPhoneNumberPager as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Pager phone number.
Notes: This label takes a string value.

7.34.11 CNLabelPhoneNumberWorkFax as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Work fax number.
Notes: This label takes a string value.

7.34.12 Constructor(value as string)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new phone number object initialized with the specified phone number string.
Notes: You should initialize this with a phone number string. This method fails when the value of string is empty.

7.34.13 copy as CNPhoneNumberMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the phone number object.

7.34.14 phoneNumberWithStringValue(p as string) as CNPhoneNumberMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new phone number object initialized with the specified phone number string.
Notes: This is a convenience class method that provides the same functionality as Constructor.
This method fails when the value of stringValue is "".

7.34.15 Properties

7.34.16 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNPhoneNumber object.
(Read and Write property)

7.34.17 stringValue as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The string value of the phone number.
Notes: (Read only property)

7.35 class CNPostalAddressFormatterMBS

7.35.1 class CNPostalAddressFormatterMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNPostalAddressFormatter class formats the postal address in a contact.
Notes: This class handles international formatting of postal addresses. It is recommended that you create an instance of this class when formatting many postal addresses, and use the instance methods; otherwise use the class methods.

7.35.2 Methods

7.35.3 attributedStringFromPostalAddress(postalAddress as CNPostalAddressMBS, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a formatted postal address as an attributed string.
Notes: postalAddress: The postal address to format.
DefaultAttributes: The default attributes to use. To learn more, see NSFormatter.

Returns the formatted postal address as an attributed string.

This method behaves similarly to stringFromPostalAddress, except that it returns an attributed string. It includes the attribute key CNPostalAddressPropertyAttribute, whose attribute values are postal address property keys, such as CNPostalAddressStreetKey. This identifies the postal address components in the formatted postal address. Also includes the attribute key CNPostalAddressLocalizedPropertyNameAttribute whose attribute values are the localized strings for the postal address property keys.
See also:

- 7.35.4 attributedStringFromPostalAddress(postalAddress as CNPostalAddressMBS, style as Integer, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS 535

7.35.4 attributedStringFromPostalAddress(postalAddress as CNPostalAddressMBS, style as Integer, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a formatted postal address as an attributed string.
Notes: postalAddress: The postal address to format.

DefaultAttributes: The default attributes to use. To learn more, see NSFormatter.

Returns the formatted postal address as an attributed string.

This method behaves similarly to stringFromPostalAddress, except that it returns an attributed string. It includes the attribute key CNPostalAddressPropertyAttribute, whose attribute values are postal address property keys, such as CNPostalAddressStreetKey. This identifies the postal address components in the formatted postal address. Also includes the attribute key CNPostalAddressLocalizedPropertyNameAttribute whose attribute values are the localized strings for the postal address property keys.
See also:

- 7.35.3 attributedStringFromPostalAddress(postalAddress as CNPostalAddressMBS, DefaultAttributes as Dictionary = nil) as NSAttributedStringMBS 535

7.35.5 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.35.6 CNPostalAddressLocalizedPropertyNameAttribute as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: This constant is a key in the attributed string whose value is a localized version of the CNPostalAddress property key.
Notes: This label takes a string value.

7.35.7 CNPostalAddressPropertyAttribute as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: This constant is a key in the attributed string whose value is a CNPostalAddress property key.
Notes: This key takes a string value.

7.35.8 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.35.9 stringFromPostalAddress(postalAddress as CNPostalAddressMBS) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a formatted postal address.
See also:

- 7.35.10 stringFromPostalAddress(postalAddress as CNPostalAddressMBS, style as Integer) as String 537

7.35.10 stringFromPostalAddress(postalAddress as CNPostalAddressMBS, style as Integer) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a formatted postal address.
Notes: Style can currently only be CNPostalAddressFormatterStyleMailingAddress.
See also:

- 7.35.9 stringFromPostalAddress(postalAddress as CNPostalAddressMBS) as String 537

7.35.11 Properties

7.35.12 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNPostalAddressFormatter object.
(Read and Write property)

7.35.13 Style as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The style to use.
Notes: Style can currently only be CNPostalAddressFormatterStyleMailingAddress.
(Read and Write property)

7.35.14 Constants

Constants

Constant	Value	Description
CNPostalAddressFormatterStyleMailingAddress	0	One of the style constants.

7.36 class CNPostalAddressMBS

7.36.1 class CNPostalAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNPostalAddress class defines an immutable object that represents the postal address for a contact.
Notes: This is a thread-safe class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Xojo Developer Magazine

- 18.5, page 83: Maps (Part 6), A deep dive into Annotations by Markus Winter

7.36.2 Methods

7.36.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.36.4 CNPostalAddressCityKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: City.
This key takes a string value.

7.36.5 CNPostalAddressCountryKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: Country.
This key takes a string value.

7.36.6 CNPostalAddressISOCountryCodeKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: ISO country code.
This key takes a string value.

7.36.7 CNPostalAddressPostalCodeKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: Postal code.
This key takes a string value.

7.36.8 CNPostalAddressStateKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: State.
This key takes a string value.

7.36.9 CNPostalAddressStreetKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties to fetch.
Notes: Street.
This key takes a string value.

7.36.10 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

7.36.11 copy as CNPostalAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the postal address object.

7.36.12 localizedStringForKey(key as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the localized name for the property associated with the specified key.
Example:

MsgBox CNPostalAddressMBS.localizedStringForKey(CNPostalAddressMBS.CNPostalAddressStreetKey)

7.36.13 mutableCopy as CNMutablePostalAddressMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a mutable copy of this postal address.

7.36.14 Properties

7.36.15 City as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The city name in a postal address.
Notes: (Read only property)

7.36.16 Country as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The country name in a postal address.
Notes: (Read only property)

7.36.17 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNPostalAddress object.
(Read and Write property)

7.36.18 ISOCountryCode as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The ISO country code for the country in a postal address.
Notes: (Read only property)

7.36.19 PostalCode as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The postal code in a postal address.
Notes: (Read only property)

7.36.20 State as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The state name in a postal address.
Notes: (Read only property)

7.36.21 Street as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The street name in a postal address.
Notes: Multiline addresses are delimited by carriage returns (that is, ,Äú\n,Äù).
(Read only property)

7.37 class CNSaveRequestMBS

7.37.1 class CNSaveRequestMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNSaveRequest class defines a save request operation for contacts.
Notes: The CNSaveRequest class creates a new save request for each save operation on the contact store. You can batch multiple changes into one save request (note that these changes only apply to objects). In the case of overlapping changes in multiple or concurrent save requests, the last change wins. If you try to add an object (that is, a contact, or a group,) that already exists in the contact store, the CNErrorCodeInsertedRecordAlreadyExists error occurs and the CNErrorUserInfoAffectedRecordsKey array is updated to contain the object you tried to add. If you try to update or delete an object that is not present in the contact store, the save request does not perform the update or deletion, the CNErrorCodeRecordDoesNotExist error occurs, and the CNErrorUserInfoAffectedRecordsKey array is updated to contain the object you tried to update or delete. Do not access objects in the save request while a save request is executing.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 22.5
- MBS Xojo Plugins, version 22.5pr4

7.37.2 Methods

7.37.3 addContact(contact as CNMutableContactMBS, ContainerIdentifier as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds the specified contact to the contact store.
Notes: contact: The new contact to add.
identifier: The identifier of the container to add the new contact. To add the new contact to the default container set identifier to "".

This method overrides any previously made deletion requests for the contact. The new contact may be modified by executing the save request.

7.37.4 addGroup(group as CNMutableGroupMBS, identifier as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a group to the contact store.
Notes: group: The group to add.
identifier: The identifier of the container to add the new group. To add the new group to the default con-

tainer, set identifier to "".

This method overrides any previously made delete request for the group.

7.37.5 addMember(contact as CNContactMBS, group as CNGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a contact as a member of a group.
Notes: contact: The contact to add to the group membership.
group: The group to add the contact to its membership.

This method overrides any previously made remove membership request on the contact from the group.

7.37.6 addSubgroup(subgroup as CNGroupMBS, group as CNGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Adds a subgroup to a group.

7.37.7 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.37.8 Constructor

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

7.37.9 deleteContact(contact as CNMutableContactMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Deletes a contact from the contact store.
Notes: contact: Contact to be delete.

7.37.10 deleteGroup(contact as CNMutableGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Deletes a group from the contact store.
Notes: group: The group to delete.

This method overrides any previously made add request on the group. The group to be deleted must already exist in the contact store. If it does not, the delete request fails, the CNErrorCodeRecordDoesNotExist error is thrown, and the CNErrorUserInfoAffectedRecordsKey array is updated to contain that object.

7.37.11 removeMember(contact as CNContactMBS, group as CNGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a contact as a member of a group.
Notes: contact: The contact to remove from the group membership.
group: The group to remove the contact from its membership.

This method removes the contact from the group, but does not delete it from the contact store. This method overrides any previously made add membership request on the contact to the group.

7.37.12 removeSubgroup(subgroup as CNGroupMBS, group as CNGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Removes a subgroup from a group.

7.37.13 updateContact(contact as CNMutableContactMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Updates an existing contact in the contact store.
Notes: contact: The contact to update.

The contact to be updated must already exist in the contact store. If it does not, the update request fails, the CNErrorCodeRecordDoesNotExist error occurs, and the CNErrorUserInfoAffectedRecordsKey array is

updated to contain the object. Note that the contact may be modified when the save request is executing.

7.37.14 updateGroup(contact as CNMutableGroupMBS)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Updates an existing group in the contact store.
Notes: group: The group to update.

The group to be updated must already exist in the contact store. If it does not, the update request fails, the CNErrorCodeRecordDoesNotExist error is thrown, and the CNErrorUserInfoAffectedRecordsKey array is updated to contain that object.

7.37.15 Properties

7.37.16 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNSaveRequest object.
(Read and Write property)

7.37.17 shouldRefetchContacts as Boolean

Plugin Version: 22.5, Platform: macOS, Targets: Desktop & iOS.

Function: Should the contacts be refetched as part of executing the save request.
Notes: Default is true where added and updated contacts are refetched by the executing save request.
Set to false to suppress this refetch behavior and reduce the execution time of the save request.

If set to false do not use the contacts after the executed save request as they may not be in a current state.
(Read and Write property)

7.37.18 transactionAuthor as String

Plugin Version: 22.0, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The author of this transaction.
Notes: Use this, in conjunction with CNChangeHistoryFetchRequestMBS.excludedTransactionAuthors, to

suppress fetching of changes the author already knows about.
(Read and Write property)

7.38 class CNSocialProfileMBS

7.38.1 class CNSocialProfileMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The CNSocialProfile class defines an immutable object representing a social profile.
Example:

```
dim ContactStore as new CNContactStoreMBS
dim c as new CNMutableContactMBS

c.givenName = "Bob"

dim sr as new CNSaveRequestMBS

dim ContainerIdentifier as string = ContactStore.defaultContainerIdentifier
sr.addContact c, ContainerIdentifier

dim e as NSErrorMBS
if ContactStore.executeSaveRequest(sr, e) then
MsgBox "Saved"
else
MsgBox "Failed to save contact"+EndOfLine+EndOfLine+e.localizedDescription
end if
```

Notes: This is a thread-safe class. Some social profile services, such as Facebook and Twitter are predefined in this class. You can also specify your own social profile service with Constructor.

7.38.2 Methods

7.38.3 available as Boolean

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.
Notes: Should return true in a 64-bit Mac app on Mac OS X 10.11 or newer.

7.38.4 CNSocialProfileServiceFacebook as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.

Notes: Facebook

7.38.5 CNSocialProfileServiceFlickr as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Flickr

7.38.6 CNSocialProfileServiceGameCenter as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Game Center

7.38.7 CNSocialProfileServiceKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties.
Notes: The social profile service.
This key takes a string value.

7.38.8 CNSocialProfileServiceLinkedIn as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: LinkedIn

7.38.9 CNSocialProfileServiceMySpace as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: MySpace

7.38.10 CNSocialProfileServiceSinaWeibo as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Sina Weibo

7.38.11 CNSocialProfileServiceTencentWeibo as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Tencent Weibo

7.38.12 CNSocialProfileServiceTwitter as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Twitter

7.38.13 CNSocialProfileServiceYelp as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the possible service keys.
Notes: Yelp

7.38.14 CNSocialProfileURLStringKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties.
Notes: The URL of the service.

7.38.15 CNSocialProfileUserIdentifierKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties.
Notes: The social profile user identifier.
This key takes a string value.

7.38.16 CNSocialProfileUsernameKey as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for properties.
Notes: The social profile user name.
This key takes a string value.

7.38.17 Constructor(URLString as String, UserName as String, Identifier as String, Service as String)

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes a new social profile object with the specified URL.
Notes: urlString: The URL for the social profile.
username: The user name for the social profile.
userIdentifier: The service,Äôs user identifier for the social profile.
service: The service name of the social profile.

7.38.18 copy as CNSocialProfileMBS

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a copy of the social profile object.

7.38.19 localizedStringForKey(key as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the localized name of the property for the specified key.
Example:

MsgBox CNSocialProfileMBS.localizedStringForService(CNSocialProfileServiceGameCenter)

Notes: key: Key for which to get the localized property name.

7.38.20 localizedStringForService(service as String) as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the localized name of the specified service.
Example:

MsgBox CNSocialProfileMBS.localizedStringForKey(CNSocialProfileUsernameKey)

Notes: service: The service name for which to get the localized name.

7.38.21 Properties

7.38.22 Handle as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: Value is a pointer to a CNSocialProfile object.
(Read and Write property)

7.38.23 service as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The social profile,Äôs service name.
Notes: (Read only property)

7.38.24 urlString as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The URL associated with the social profile.
Notes: (Read only property)

7.38.25 userIdentifier as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The service,Äôs user identifier associated with the social profile.
Notes: (Read only property)

7.38.26 username as String

Plugin Version: 16.3, Platform: macOS, Targets: Desktop & iOS.

Function: The user name for the social profile.
Notes: (Read only property)

Chapter 8

CoreML

8.1 class MLArrayBatchProviderMBS

8.1.1 class MLArrayBatchProviderMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A convenience wrapper for batches of feature providers.
Notes: This batch provider supports an array of feature providers or a dictionary of arrays of feature values.
Available on MacOS 10.14 or newer.
Subclass of the MLBatchProviderMBS class.
Blog Entries

- Update Machine Learning Model on Device
- MBS Xojo Plugins, version 19.6pr3

8.1.2 Methods

8.1.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.14 or newer.

8.1.4 Constructor(value as Dictionary, byref Error as NSErrorMBS)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a batch provider based on feature names and their associated arrays of data.
Notes: dictionary: A dictionary which maps feature names to an array of values. The error case occurs when all the arrays do not have the same length or the values in an aray are not expressible as an MLFeatureValueMBS.
See also:

- 8.1.5 Constructor(values() as MLFeatureProviderMBS) 556

8.1.5 Constructor(values() as MLFeatureProviderMBS)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates the batch provider based on the array of feature providers.
Notes: array: The array of feature providers for the batch.
See also:

- 8.1.4 Constructor(value as Dictionary, byref Error as NSErrorMBS) 555

8.1.6 values as MLFeatureProviderMBS()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The array of feature providers.

8.2 class MLBatchProviderMBS

8.2.1 class MLBatchProviderMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: An interface that represents a collection of feature providers.
Notes: Similar to the MLFeatureProviderMBS, this interface allows you to define your own batch provider. If you collect your data asynchronously or it is memory intensive, implement this protocol on your data structure to optimize performance with batch processing.

Available on MacOS 10.14 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.2.2 Methods

8.2.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.2.4 featuresAtIndex(index as Integer) as MLFeatureProviderMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Returns the feature provider at the given index.

8.2.5 Properties

8.2.6 Count as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The number of feature providers in this batch.
Notes: (Read only property)

8.2.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.3 class MLDictionaryConstraintMBS

8.3.1 class MLDictionaryConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint describing expected NSDictionary properties.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.3.2 Methods

8.3.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.13 or newer.

8.3.4 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.3.5 Properties

8.3.6 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.3.7 keyType as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Required key type.
Notes: See MLFeatureValueMBS.Type* constants.

(Read only property)

8.4 class MLDictionaryFeatureProviderMBS

8.4.1 class MLDictionaryFeatureProviderMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: A concrete convenience class conforming to MLFeatureProvider protocol.
Notes: Subclass of the MLFeatureProviderMBS class.
Blog Entries

- Update Machine Learning Model on Device

8.4.2 Methods

8.4.3 Constructor(content as Dictionary, byref error as NSErrorMBS)

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Create from a generic dictionary by converting all values to MLFeatureValues or from a dictionary with values already stored as MLFeatureValues.
Notes: An error results if the values are not or cannot be represented as MLFeatureValues.

8.4.4 objectForKeyedSubscript(script as string) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Get the value for specified feature.

8.4.5 Properties

8.4.6 Content as Dictionary

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Dictionary holding the feature values.
Notes: When you query this, you get a copy of the dictionary, so any modification doesn't go back to the feature provider.
(Read only property)

8.5 class MLFeatureDescriptionMBS

8.5.1 class MLFeatureDescriptionMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Description of a feature.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.5.2 Methods

8.5.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.5.4 copy as MLFeatureDescriptionMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Creates a copy.

8.5.5 isAllowedValue(value as MLFeatureValueMBS) as boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Check if MLFeatureValue is valid based on this description.

8.5.6 Properties

8.5.7 dictionaryConstraint as MLDictionaryConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint for dictionary.
Notes: Only set when type is dictionary.
(Read only property)

8.5.8 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.5.9 imageConstraint as MLImageConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint for image.
Notes: Only set when type is image.
(Read only property)

8.5.10 isOptional as Boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Whether this feature can take an undefined value or not.
Notes: (Read only property)

8.5.11 multiArrayConstraint as MLMultiArrayConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint for multi array.
Notes: Only set when type is array.
(Read only property)

8.5.12 Name as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Name of feature.
Notes: (Read only property)

8.5.13 sequenceConstraint as MLSequenceConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constraints for a sequence feature.
Notes: (Read only property)

8.5.14 Type as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Type of data.
Notes: (Read only property)

8.5.15 Constants

Value Types

Constant	Value	Description
TypeDictionary	6	Numerically weighted hashable objects (e.g. word counts)
TypeDouble	2	Continuous values
TypeImage	4	CVPixelBufferRef or converted by plugin to Xojo picture.
TypeInt64	1	Discrete values, sometimes used to hold numeric encoding of a categorical value
TypeInvalid	0	Undefined type.
TypeMultiArray	5	An array of values.
TypeSequence	7	The possible types for feature values, input features, and output features.
TypeString	3	Text or categorical strings

8.6 class MLFeatureProviderMBS

8.6.1 class MLFeatureProviderMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Protocol for accessing a feature value for a feature name.
Notes: In Xojo defined as class, but in objective-c just a protocol.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.6.2 Methods

8.6.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.6.4 featureNames as String()

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Queries list of all feature names available.

8.6.5 featureValueForName(featureName as String) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Queries value for feature by name.
Notes: Returns nil if the provided featureName is not in the set of featureNames

8.6.6 Properties

8.6.7 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.7 class MLFeatureValueMBS

8.7.1 class MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: An immutable variant holding a data value of a supported MLFeatureType.
Notes: MLFeatureValue does not support type conversion in its accessor properties.
It can also have a missing or undefined value of a well defined type.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 21.5pr2
- Update Machine Learning Model on Device

8.7.2 Methods

8.7.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.7.4 copy as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Creates a copy.

8.7.5 featureValueWithCGImage(image as variant, orientation as integer = -1, constraint as MLImageConstraintMBS, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps an image as a value for a feature, given an image,Äôs URL and constraints.
Notes: Available on macOS 10.15 or newer.
If orientation is in range 1 to 8, we pass that as orientation. Otherwise orientation will come from EXIF
See also:

- 8.7.6 featureValueWithCGImage(image as variant, orientation as integer = -1, pixelsWide as Integer, pixelsHigh as Integer, pixelFormatType as String, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS 567

8.7.6 featureValueWithCGImage(image as variant, orientation as integer = -1, pixelsWide as Integer, pixelsHigh as Integer, pixelFormatType as String, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps an image as a value for a feature, given an image, its width, and its height, in pixels.
Notes: Available on macOS 10.15 or newer.
If orientation is in range 1 to 8, we pass that as orientation. Otherwise orientation will come from EXIF
See also:

- 8.7.5 featureValueWithCGImage(image as variant, orientation as integer = -1, constraint as MLImageConstraintMBS, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS 566

8.7.7 featureValueWithDictionary(value as Dictionary, byref error as NSErrorMBS) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Creates new feature based on a dictionary.
Notes: Copies the dictionary.
For encoding a sparse feature set or for encoding probabilities. Input keys that are not number or string are rejected on construction and return a MLModelErrorFeatureTypeMismatch error. Further validation for consistency occurs on evaluation.

8.7.8 featureValueWithDouble(value as double) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Returns feature object for given value.

8.7.9 featureValueWithImageFile(File as FolderItem, orientation as integer = -1, constraint as MLImageConstraintMBS, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps an image as a value for a feature, given an image and its constraints.
Notes: Available on macOS 10.15 or newer.
If orientation is in range 1 to 8, we pass that as orientation. Otherwise orientation will come from EXIF
See also:

- 8.7.10 featureValueWithImageFile(File as FolderItem, orientation as integer = -1, pixelsWide as Integer, pixelsHigh as Integer, pixelFormatType as String, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS 568

8.7.10 featureValueWithImageFile(File as FolderItem, orientation as integer = -1, pixelsWide as Integer, pixelsHigh as Integer, pixelFormatType as String, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps an image as a value for a feature, given an image,Äôs URL, and its width and height, in pixels.
Notes: Available on macOS 10.15 or newer.
If orientation is in range 1 to 8, we pass that as orientation. Otherwise orientation will come from EXIF
See also:

- 8.7.9 featureValueWithImageFile(File as FolderItem, orientation as integer = -1, constraint as MLImageConstraintMBS, options as Dictionary = nil, byref error as NSErrorMBS) as MLFeatureValueMBS 567

8.7.11 featureValueWithInt64(value as Int64) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Returns feature object for given value.

8.7.12 featureValueWithMultiArray(value as MLMultiArrayMBS) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Returns feature object for given value.

8.7.13 featureValueWithPicture(value as Picture) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Returns feature object for given value.

8.7.14 featureValueWithPixelBuffer(Handle as Integer) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps a pixel buffer as a value for a feature.

8.7.15 featureValueWithSequence(sequence as MLSequenceMBS) as MLFeatureValueMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Wraps a sequence as a value for a feature.

8.7.16 featureValueWithString(value as string) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Returns feature object for given value.

8.7.17 isEqualToFeatureValue(value as MLFeatureValueMBS) as Boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Queries whether two feature objects are the same.

8.7.18 MLFeatureValueImageOptionCropAndScale as String

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The option you use to crop and scale an image when creating an image feature value.

8.7.19 MLFeatureValueImageOptionCropRect as String

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The option you use to crop an image when creating an image feature value.

8.7.20 undefinedFeatureValueWithType(type as Integer) as MLFeatureValueMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Represent an undefined value of a specified type.

8.7.21 Properties

8.7.22 CIImageValue as Variant

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Queries image as CIImageMBS object.
Notes: (Read only property)

8.7.23 dictionaryValue as Dictionary

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeDictionary.
Notes: When you query the dictionary, you get a copy.
(Read only property)

8.7.24 doubleValue as Double

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeDouble.
Notes: (Read only property)

8.7.25 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.7.26 imageBufferValue as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The image, or pixelBuffer, wrapped in this feature value.
Notes: The handle for a CVPixelBufferMBS object.
(Read only property)

8.7.27 int64Value as Int64

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeInt64.
Notes: (Read only property)

8.7.28 multiArrayValue as MLMultiArrayMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeMultiArray.
Notes: (Read only property)

8.7.29 PictureHeight as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The picture height.
Notes: Populated value if the type is TypeImage.
(Read only property)

8.7.30 PictureValue as Picture

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeImage.
Notes: (Read only property)

8.7.31 PictureWidth as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The picture width.
Notes: Populated value if the type is TypeImage.
(Read only property)

8.7.32 sequenceValue as MLSequenceMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The sequence wrapped in this feature value.
Notes: (Read only property)

8.7.33 stringValue as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Populated value if the type is TypeString.
Notes: (Read only property)

8.7.34 Type as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Type of the value for which the corresponding property below is held.
Notes: (Read only property)

8.7.35 Undefined as Boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: True if the value represents a missing or undefined value.
Notes: (Read only property)

8.7.36 value as Variant

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Queries value as variant.
Notes: (Read only property)

8.7.37 Constants

Value Types

Constant	Value	Description
TypeDictionary	6	Numerically weighted hashable objects (e.g. word counts)
TypeDouble	2	Continuous values
TypeImage	4	CVPixelBufferRef or converted by plugin to Xojo picture.
TypeInt64	1	Discrete values, sometimes used to hold numeric encoding of a categorical value
TypeInvalid	0	Undefined type.
TypeMultiArray	5	An array of values.
TypeSequence	7	The possible types for feature values, input features, and output features.
TypeString	3	Text or categorical strings

8.8 class MLImageConstraintMBS

8.8.1 class MLImageConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint on image properties.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.8.2 Methods

8.8.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.8.4 Properties

8.8.5 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.8.6 pixelFormatType as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The accepted kCVPixelFormatType for the image.
Notes: (Read only property)

8.8.7 pixelsHigh as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The required height of the image.
Notes: (Read only property)

8.8.8 pixelsWide as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The required width of the image.
Notes: (Read only property)

8.8.9 sizeConstraint as MLImageSizeConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Additional sizes this image feature supports.
Notes: Available on MacOS 10.14 or newer.
(Read only property)

8.9 class MLImageSizeConstraintMBS

8.9.1 class MLImageSizeConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A list or range of sizes that augment an image constraint's default size.
Notes: You use an MLImageSizeConstraint to express what image sizes of an image feature a model will accept as input or produce as output.

Use type to determine which properties describe what image sizes the model's image feature expects as input or produces as output.

If type is:

- TypeRange, the image feature accepts any image that has a width in pixelsWideRange and a height in pixelsHighRange.
- TypeEnumerated, the image feature accepts any image size listed in enumeratedImageSizes.
- TypeUnspecified, the MLImageSizeConstraint instance is not configured and should be ignored. Instead, use the image feature's default image size constraint, defined by pixelsWide and pixelsHigh.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.9.2 Methods

8.9.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.14 or newer.

8.9.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.9.5 enumeratedImageSizes as MLImageSizeMBS()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: An array of image sizes a model's image feature accepts as input or produces as output.

8.9.6 Properties

8.9.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.9.8 pixelsHighRange as NSRangeMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The range of heights a model's image feature accepts as input or produces as output.
Notes: (Read only property)

8.9.9 pixelsWideRange as NSRangeMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The range of widths a model's image feature accepts as input or produces as output.
Notes: (Read only property)

8.9.10 type as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Indicator of which properties to inspect for this image size constraint.
Notes: (Read only property)

8.9.11 Constants

Types

Constant	Value	Description
TypeEnumerated	2	The image feature accepts image sizes listed in an array.
TypeRange	3	The image feature accepts image sizes defined by a range of widths and a range of heights.
TypeUnspecified	0	The image size constraint is not configured and should be ignored.

8.10 class MLImageSizeMBS

8.10.1 class MLImageSizeMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The width and height of an image feature size.
Notes: Available on MacOS 10.14 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.10.2 Methods

8.10.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.14 or newer.

8.10.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.10.5 Properties

8.10.6 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.10.7 pixelsHigh as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The height of an image feature in pixels.
Notes: (Read only property)

8.10.8 pixelsWide as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The width of an image feature in pixels.
Notes: (Read only property)

8.11 class MLKeyMBS

8.11.1 class MLKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: An abstract base class for machine learning key types.
Notes: You don,Äôt create use this class directly. Instead, use a class that inherits from this one, such as MLParameterKey or MLMetricKey.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.11.2 Methods

8.11.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.11.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.11.5 copy as MLKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Copies the object.

8.11.6 Properties

8.11.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.11.8 Name as String

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The name of the machine learning key.
Notes: (Read only property)

8.11.9 Scope as String

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The scope value.
Notes: (Read only property)

8.12 class MLMetricKeyMBS

8.12.1 class MLMetricKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A key for the metrics dictionary in an update context.
Notes: Available in MacOS 10.15 or newer.
Subclass of the MLKeyMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.12.2 Methods

8.12.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.12.4 Properties

8.12.5 epochIndex as MLMetricKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the epoch index (an Int64 value).
Notes: Use this key to fetch the epoch index value in the metrics dictionary.
(Read only property)

8.12.6 lossValue as MLMetricKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the current loss (a float value).
Notes: Use this key to fetch the loss value in the metrics dictionary.
(Read only property)

8.12.7 miniBatchIndex as MLMetricKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the mini-batch index (an Int64 value) within an epoch.
Notes: Use this key to fetch the mini-batch index value in the metrics dictionary.
(Read only property)

8.13 class MLModelConfigurationMBS

8.13.1 class MLModelConfigurationMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The settings for creating or updating a machine learning model.
Notes: Available in MacOS 10.14 or newer.
Blog Entries

- Update Machine Learning Model on Device
- MBS Xojo Plugins, version 19.6pr3

8.13.2 Methods

8.13.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.14 or newer.

8.13.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constructor.

8.13.5 copy as MLModelConfigurationMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a copy of the model configuration.

8.13.6 Properties

8.13.7 allowLowPrecisionAccumulationOnGPU as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A Boolean value that determines whether to allow low-precision accumulation on a GPU.
Notes: (Read and Write property)

8.13.8 computeUnits as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The processing unit or units the model uses to make predictions.
Notes: (Read and Write property)

8.13.9 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.13.10 parameters as Dictionary

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Optional dictionary with any requested changes to model or update parameters.
Notes: (Read and Write property)

8.13.11 preferredMetalDevice as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The metal device you prefer this model use to make predictions (inference) and update the model.
Notes: If preferredMetalDevice is 0, the default value, Core ML chooses a metal device for you.
Until we have a class for MLDevice, you pass the handle here directly.
(Read and Write property)

8.13.12 Constants

Compute Units

Constant	Value	Description
ComputeUnitsAll	2	The option you choose to allow the model to use all compute units available, including the neural engine.
ComputeUnitsCPUAndGPU	1	The option you choose to allow the model to use both the CPU and GPU, but not the neural engine.
ComputeUnitsCPUOnly	0	The option you choose to limit the model to only use the CPU.

8.14 class MLModelDescriptionMBS

8.14.1 class MLModelDescriptionMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: A description of a model containing input and output feature descriptions, optionally outputted features with special meaning and metadata.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.14.2 Methods

8.14.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.14.4 Properties

8.14.5 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.14.6 inputDescriptionsByName as Dictionary

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Description of the inputs to the model.
Notes: (Read only property)

8.14.7 isUpdatable as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether you can update the model with additional training.
Notes: Available on Mac OS 10.15 or newer.
(Read only property)

8.14.8 metadata as Dictionary

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Optional metadata describing the model.
Notes: (Read only property)

8.14.9 outputDescriptionsByName as Dictionary

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Description of the outputs from the model.
Notes: (Read only property)

8.14.10 parameterDescriptionsByKey as Dictionary

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Allows for access of each parameter as parameter description.
Notes: (Read only property)

8.14.11 predictedFeatureName as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Name of the primary target / predicted output feature in the output descriptions.
Notes: (Read only property)

8.14.12 predictedProbabilitiesName as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Key for all predicted probabilities stored as a MLFeatureTypeDictionary in the output descriptions.
Notes: (Read only property)

8.14.13 trainingInputDescriptionsByName as Dictionary

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Provides access to each training input as a feature description, given the name of the input.
Notes: Available on MacOS 101.5 or newer.
(Read only property)

8.15 class MLModelMBS

8.15.1 class MLModelMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Construct a model and evaluate on a specific set of input features.
Notes: Inputs and outputs are accessed via the MLFeatureProvider protocol.
Returns a model or nil if there is an error.

Available on macOS 10.13 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Multithreaded plugin functions can increase speed of Xojo application
- Update Machine Learning Model on Device

8.15.2 Methods

8.15.3 available as Boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Whether CoreML is available.
Notes: Should return true on macOS 10.13 or newer.

8.15.4 compileModelAtURL(URL as string, byref error as NSErrorMBS) as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Compile a .mlmodel for this device.
Notes: URL: URL file path to .mlmodel file you wish to compile
error: Any errors are surfaced here

Returns a URL to the compiled .mlmodelc bundle if successful.
The model is compiled to a temporary location on disk.
You must move the compiled model to a permenant location if you wish to keep it.

8.15.5 compileModelFile(File as folderItem, byref error as NSErrorMBS) as folderItem

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Compile a .mlmodel for this device.
Example:

```
dim SourceFile as FolderItem = SpecialFolder.Desktop.child("resnet50.MLmodel")
dim Error as NSErrorMBS

dim Tempfile as FolderItem = MLModelMBS.compileModelFile(SourceFile, Error)

if Tempfile <>nil then
tempfile.MoveFileTo SpecialFolder.Desktop
MsgBox "OK"
else
MsgBox Error.localizedDescription
end if
```

Notes: File: FolderItem for .mlmodel file you wish to compile
error: Any errors are surfaced here

Returns a folderitem to the compiled .mlmodelc bundle if successful.
The model is compiled to a temporary location on disk.
You must move the compiled model to a permenant location if you wish to keep it.

8.15.6 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.15.7 MLModelAuthorKey as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: One of the keys for the metadata dictionary.
Notes: The author of this model.

8.15.8 MLModelCreatorDefinedKey as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: One of the keys for the metadata dictionary.
Notes: Any additional pertinent information specified by the model creator.

8.15.9 MLModelDescriptionKey as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: One of the keys for the metadata dictionary.
Notes: A short description of what the model does and/or its purpose

8.15.10 MLModelErrorDomain as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The error domain string for the CoreML errors.

8.15.11 MLModelLicenseKey as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: One of the keys for the metadata dictionary.
Notes: License information for the model.

8.15.12 MLModelVersionStringKey as String

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: One of the keys for the metadata dictionary.
Notes: A version number encoded as a string.

8.15.13 modelWithContentsOfFile(file as FolderItem, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Construct a model given the location of its on-disk representation.
Notes: Returns nil on error.
See also:

- 8.15.14 modelWithContentsOfFile(file as FolderItem, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS 594

8.15.14 modelWithContentsOfFile(file as FolderItem, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a Core ML model object from a compiled model file and a custom configuration.
Notes: File: The path to a compiled model file (ModelName.mlmodelc), typically with the URL returned from compileModelAtURL:error:.
configuration: The model settings for a new model object.

However, if the wrapper class doesn't meet your app's needs or you need to customize the model's configuration, then use this initializer to create a model object from any compiled model file your app has access to. Typically, you use this initializer after your app has downloaded and compiled a model, which is one technique for saving space in your app.
See also:

- 8.15.13 modelWithContentsOfFile(file as FolderItem, byref error as NSErrorMBS) as MLModelMBS 593

8.15.15 modelWithContentsOfPath(Path as string, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Construct a model given the location of its on-disk representation.
Notes: Returns nil on error.
See also:

- 8.15.16 modelWithContentsOfPath(Path as string, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS 594

8.15.16 modelWithContentsOfPath(Path as string, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a Core ML model object from a compiled model file and a custom configuration.
Notes: Path: The path to a compiled model file (ModelName.mlmodelc), typically with the URL returned

from compileModelAtURL:error:.
configuration: The model settings for a new model object.

However, if the wrapper class doesn't meet your app's needs or you need to customize the model's configuration, then use this initializer to create a model object from any compiled model file your app has access to. Typically, you use this initializer after your app has downloaded and compiled a model, which is one technique for saving space in your app.
See also:

- 8.15.15 modelWithContentsOfPath(Path as string, byref error as NSErrorMBS) as MLModelMBS 594

8.15.17 modelWithContentsOfURL(URL as string, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Construct a model given the location of its on-disk representation.
Notes: Returns nil on error.
See also:

- 8.15.18 modelWithContentsOfURL(URL as string, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS 595

8.15.18 modelWithContentsOfURL(URL as string, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) as MLModelMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a Core ML model object from a compiled model file and a custom configuration.
Notes: URL: The path to a compiled model file (ModelName.mlmodelc), typically with the URL returned from compileModelAtURL:error:.
configuration: The model settings for a new model object.

However, if the wrapper class doesn't meet your app's needs or you need to customize the model's configuration, then use this initializer to create a model object from any compiled model file your app has access to. Typically, you use this initializer after your app has downloaded and compiled a model, which is one technique for saving space in your app.
See also:

- 8.15.17 modelWithContentsOfURL(URL as string, byref error as NSErrorMBS) as MLModelMBS 595

8.15.19 parameterValueForKey(key as MLParameterKeyMBS, byref error as NSErrorMBS) as Variant

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Provides value for the given parameter.
Notes: Returns nil on error.
Available on MacOS 10.15 or newer.

8.15.20 predictionFromFeatures(input as MLFeatureProviderMBS, options as MLPredictionOptionsMBS = nil, byref error as NSErrorMBS) as MLFeatureProviderMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: All models can predict on a specific set of input features.

8.15.21 predictionFromFeaturesMT(input as MLFeatureProviderMBS, options as MLPredictionOptionsMBS = nil, byref error as NSErrorMBS) as MLFeatureProviderMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: All models can predict on a specific set of input features.
Notes: The work is performed on an extra thread, so this function can yield time to other Xojo threads. And it calles the Working event regularly.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.
If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

8.15.22 predictionsFromBatch(inputBatch as MLBatchProviderMBS, byref error as NSErrorMBS) as MLBatchProviderMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Predicts output feature values from the given a batch of input feature values.
Notes: inputBatch: The batch of feature values the model needs to make its predictions.

Returns a batch provider that represents the model,Äôs predictions for the batch of inputs.
See also:

8.15.23 predictionsFromBatch(inputBatch as MLBatchProviderMBS, options as MLPredictionOptionsMBS, byref error as NSErrorMBS) as MLBatchProviderMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Predicts output feature values from the given a batch of input feature values and prediction options.
Notes: inputBatch: The batch of feature values the model needs to make its predictions.
options: All the options to be applied to the prediction.

Returns a batch provider that represents the model,Äôs predictions for the batch of inputs.
See also:

8.15.24 Properties

8.15.25 configuration as MLModelConfigurationMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The configuration of the model set during initialization.
Notes: (Read only property)

8.15.26 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.15.27 modelDescription as MLModelDescriptionMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: A model holds a description of its required inputs and expected outputs.
Notes: (Read only property)

8.15.28 Constants

Errors

Constant	Value	Description
ErrorCustomLayer	4	An error related to a custom layer in your model.
ErrorCustomModel	5	An error related to your custom model.
ErrorDescriptionMismatch	2	Mismatch error.
ErrorFeatureType	1	Type error.
ErrorGeneric	0	Generic error.
ErrorIO	3	I/O file error.
ErrorParameters	7	An error related to missing or invalid model parameters.
ErrorUpdate	6	An error related to updating a model.

8.16 class MLMultiArrayConstraintMBS

8.16.1 class MLMultiArrayConstraintMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Constraint describing expected MLMultiArray properties.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.16.2 Methods

8.16.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The private constructor.

8.16.4 shape as Integer()

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Required shape of array.
Notes: One value for each dimension.

8.16.5 Properties

8.16.6 dataType as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Required dataType.
Notes: (Read only property)

8.16.7 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.16.8 shape0 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(0) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.16.9 shape1 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(1) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.16.10 shape2 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(2) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.16.11 shapeConstraint as MLMultiArrayShapeConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constraint on the shape of the multiarray.
Notes: (Read only property)

8.17 class MLMultiArrayMBS

8.17.1 class MLMultiArrayMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The class for a Multidimensional Array.
Blog Entries

- MBS Xojo Plugins, version 21.5pr4

8.17.2 Methods

8.17.3 Constructor(dataPointer as Ptr, shape() as Integer, dataType as Integer, strides() as Integer, byref error as NSErrorMBS)

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Create by wrapping existing data.
Notes: Please make sure that memory is not deallocated too early, e.g. by subclassing and freeing it in destructor.
See also:

- 8.17.4 Constructor(shape() as Integer, dataType as Integer, byref error as NSErrorMBS) 601

8.17.4 Constructor(shape() as Integer, dataType as Integer, byref error as NSErrorMBS)

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Create by C-style contiguous array by allocating and managing the necessary memory.
See also:

- 8.17.3 Constructor(dataPointer as Ptr, shape() as Integer, dataType as Integer, strides() as Integer, byref error as NSErrorMBS) 601

8.17.5 shape as Integer()

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: An array containing the sizes of each dimension in the multiarray.

8.17.6 strides as Integer()

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: An array containing the stride in memory for each dimension.
Notes: The element referred to by a multidimensional index is located at an offset equal to sum_d index [d] *strides [d] . This offset is in the units of the specified dataType.

8.17.7 Properties

8.17.8 count as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Count of total number of elements.
Notes: (Read only property)

8.17.9 dataPointer as Ptr

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Unsafe pointer to underlying buffer holding the data.
Notes: (Read only property)

8.17.10 dataType as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Type of element held.
Notes: See DataType* constants.
(Read only property)

8.17.11 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.17.12 shape0 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(0) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.13 shape1 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(1) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.14 shape2 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The shape(2) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.15 strides0 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The strides(0) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.16 strides1 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The strides(1) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.17 strides2 as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The strides(2) value.
Notes: Convenience property to see value in debugger.
(Read only property)

8.17.18 doubleValue(index as Integer) as Double

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for single dimensional array.
Notes: Pass the index for the value to query.
(Read and Write computed property)
See also:

- 8.17.19 doubleValue(indexes() as Integer) as Double 604

8.17.19 doubleValue(indexes() as Integer) as Double

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for multi dimensional array.
Notes: Pass one index value for each dimension in the array.
(Read and Write computed property)
See also:

- 8.17.18 doubleValue(index as Integer) as Double 604

8.17.20 integerValue(index as Integer) as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for single dimensional array.
Notes: Pass the index for the value to query.
(Read and Write computed property)
See also:

- 8.17.21 integerValue(indexes() as Integer) as Integer 604

8.17.21 integerValue(indexes() as Integer) as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for multi dimensional array.
Notes: Pass one index value for each dimension in the array.
(Read and Write computed property)
See also:

- 8.17.20 integerValue(index as Integer) as Integer 604

8.17.22 singleValue(index as Integer) as Single

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for single dimensional array.
Notes: Pass the index for the value to query.
(Read and Write computed property)
See also:

- 8.17.23 singleValue(indexes() as Integer) as Single 605

8.17.23 singleValue(indexes() as Integer) as Single

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Query or set value for multi dimensional array.
Notes: Pass one index value for each dimension in the array.
(Read and Write computed property)
See also:

- 8.17.22 singleValue(index as Integer) as Single 605

8.17.24 Constants

Data Types

Constant	Value	Description
DataTypeDouble	&h10040	Data is double values.
DataTypeFloat16	&h10010	Data is Float16 values. Sadly Xojo doesn't have a native Float16 data type.
DataTypeFloat32	&h10020	Data is single values (32-bit).
DataTypeFloat64	&h10040	Data is Float64 values. Use double in Xojo.
DataTypeInt32	&h20020	Data is 32-bit integer values.

8.18 class MLMultiArrayShapeConstraintMBS

8.18.1 class MLMultiArrayShapeConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The lists of shapes or ranges of shapes that constrain a multiarray feature.
Notes: Available in MacOS 10.14 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.18.2 Methods

8.18.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.18.4 enumeratedShapes as Variant

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Array of allowed shapes for a multiarray feature.

8.18.5 sizeRangeForDimension as NSRangeMBS()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The allowable range for a dimention of the multiarray.

8.18.6 Properties

8.18.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.18.8 type as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The type of the shape constraint.
Notes: (Read only property)

8.18.9 Constants

Types

Constant	Value	Description
TypeEnumerated	2	The constraint is an array of allowed shapes.
TypeRange	3	The constraint is a set of ranges allowed for the array shape.
TypeUnspecified	1	The constraint type is undefined.

8.19 class MLNumericConstraintMBS

8.19.1 class MLNumericConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The value limitations of a number.
Notes: Available on MacOS 10.15 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.19.2 Methods

8.19.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.19.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.19.5 enumeratedNumbers as Double()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A set of the numbers allowed in this constraint.

8.19.6 Properties

8.19.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.19.8 maxNumber as Double

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The largest numerical value allowed by this constraint.
Notes: (Read only property)

8.19.9 minNumber as Double

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The smallest numerical value allowed by this constraint.
Notes: (Read only property)

8.20 class MLParameterDescriptionMBS

8.20.1 class MLParameterDescriptionMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The description of a model parameter that includes a default value and a constraint, if applicable.
Notes: Available for MacOS 10.15 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.20.2 Methods

8.20.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.20.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.20.5 Properties

8.20.6 defaultValue as Variant

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The default value for the parameter.
Notes: (Read only property)

8.20.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.20.8 key as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key for this parameter description value.
Notes: (Read only property)

8.20.9 numericConstraint as MLNumericConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constraints of this paramter description value, if and only if the value is numerical.
Notes: (Read only property)

8.21 class MLParameterKeyMBS

8.21.1 class MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A key for model and update parameter dictionaries.
Notes: Subclass of the MLKeyMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.21.2 Methods

8.21.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.21.4 scopedTo(scope as String) as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Returns a new MLParameterKey instance after adding additional scoping.

8.21.5 Properties

8.21.6 beta1 as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the beta1 parameter for the Adam optimizer (a Double value).
Notes: Double parameter used to control the beta1 of Adam optimizer. Adjustable at load-time.
(Read only property)

8.21.7 beta2 as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the beta2 parameter for the Adam optimizer (a Double value).
Notes: Double parameter used to control the beta2 of Adam optimizer. Adjustable at load-time.
(Read only property)

8.21.8 biases as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: MLMultiArrayMBS parameter returned when client requests for biases of a particular layer using a scoped parameter.
Notes: (Read only property)

8.21.9 epochs as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the epochs parameter for the optimizer (an Int64 value).
Notes: Int64 parameter used to specify the number of epochs used by optimizer. Adjustable at load-time.
(Read only property)

8.21.10 eps as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the eps parameter for the Adam optimizer (a Double value).
Notes: Double parameter used to control the epsilon of Adam optimizer. Adjustable at load-time.
(Read only property)

8.21.11 learningRate as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the parameter for the optimizer,Äôs learning rate (a Double value).
Notes: Double parameter used to control the learning rate of an optimizer. Adjustable in progress.
(Read only property)

8.21.12 linkedModelFileName as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the name of the linked model.
Notes: (Read only property)

8.21.13 linkedModelSearchPath as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the search path for the linked model.
Notes: (Read only property)

8.21.14 miniBatchSize as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the mini batch-size parameter for the optimizer (an Int64 value).
Notes: Int64 parameter used to specify the size of a miniBatch used by optimizer. Adjustable at load-time.
(Read only property)

8.21.15 momentum as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The key you use to access the parameter for the optimizer,Äôs momentum gradient (a Double value).
Notes: Double parameter used to control the momentum of gradient based optimizers. Adjustable at load-time.
(Read only property)

8.21.16 numberOfNeighbors as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Int64 parameter used to specify the number of neighbors to use for class affinity (applicable to kNN).
Notes: Not adjustable in progress.
(Read only property)

8.21.17 seed as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Int64 parameter used to specify the seed to be used if shuffling data between epochs.
Notes: Adjustable at load-time.
(Read only property)

8.21.18 shuffle as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Bool parameter used to specify whether to shuffle the data between epochs.
Notes: Adjustable at load-time.
(Read only property)

8.21.19 weights as MLParameterKeyMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: MLMultiArrayMBS parameter returned when client requests for weights of a particular layer using a scoped parameter.
Notes: (Read only property)

8.22 class MLPredictionOptionsMBS

8.22.1 class MLPredictionOptionsMBS

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: An object to hold options / controls / parameters of how model prediction is performed

8.22.2 Methods

8.22.3 Constructor

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The constructor.

8.22.4 Properties

8.22.5 Handle as Integer

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.22.6 usesCPUOnly as Boolean

Plugin Version: 17.4, Platform: macOS, Targets: All.

Function: Set to true to force computation to be on the CPU only
Notes: (Read and Write property)

8.23 class MLSequenceConstraintMBS

8.23.1 class MLSequenceConstraintMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constraints for a sequence feature.
Notes: Available on MacOS 10.14 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.23.2 Methods

8.23.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.14 or newer.

8.23.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.23.5 Properties

8.23.6 countRange as NSRangeMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The range of values allowed for the sequence's length.
Notes: (Read only property)

8.23.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.23.8 valueDescription as MLFeatureDescriptionMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The description that all sequence elements must match.
Notes: (Read only property)

8.24 class MLSequenceMBS

8.24.1 class MLSequenceMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A sequence of data used as a feature input or feature output for a model.
Notes: Data from sensors (a time series) or language (a series of words) are often manipulated as sequences by machine learning models. These sequences can have variable lengths depending on their context. MLSequenceMBS explicitly encodes a sequence of strings or integers to support a series of data points as a single input.

If you need to process a sequence of values individually, one at a time, see Making Predictions with a Sequence of Inputs for details.
https://developer.apple.com/documentation/coreml/core_ml_api/making_predictions_with_a_sequence_of_inputs

Available in MacOS 10.14 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.24.2 Methods

8.24.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.24.4 emptySequenceWithType(type as Integer) as MLSequenceMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates an empty sequence of the given type.
Notes: Type: The MLFeatureType for the sequence. Must be either MLFeatureTypeString or MLFeatureTypeInt64.

8.24.5 int64Values as Int64()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The integer elements of the sequence.

8.24.6 sequenceWithInt64Array(int64Values() as Int64) as MLSequenceMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a sequence from an array of integers.

8.24.7 sequenceWithStringArray(stringValues() as String) as MLSequenceMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a sequence from an array of strings.

8.24.8 stringValues as String()

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The string elements of the sequence.

8.24.9 Properties

8.24.10 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.24.11 type as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The type of value held in the sequence.
Notes: (Read only property)

8.25 class MLTaskMBS

8.25.1 class MLTaskMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: An abstract base class for machine learning tasks.
Notes: You don,Äôt create use this class directly. Instead, use a class that inherits from this one, such as MLUpdateTaskMBS.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.6pr3

8.25.2 Methods

8.25.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.25.4 cancel

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Cancels a machine learning task before it completes.

8.25.5 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The private constructor.

8.25.6 resume

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Begins or resumes a machine learning task.
Notes: Use this method to start a task for the first time or resumes a task that has paused. Tasks pause

when they notify your app,Äôs progress handlers, such as those you provide to an MLUpdateProgressHandlersMBS instance.

8.25.7 Properties

8.25.8 Error as NSErrorMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The underlying error if the task is in a failed state.
Notes: (Read only property)

8.25.9 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.25.10 State as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The current state of the machine learning task.
Notes: (Read only property)

8.25.11 TaskIdentifier as String

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A unique name of the task to distinguish it from all other tasks at runtime.
Notes: (Read only property)

8.25.12 Constants

Task States

Constant	Value	Description
StateCancelling	3	The state of a machine learning task that,Äôs in mid-termination, before it could finish successfully.
StateCompleted	4	The state of a machine learning task that has finished successfully.
StateFailed	5	The state of a machine learning task that has terminated due to an error.
StateRunning	2	The state of a machine learning task that,Äôs executing.
StateSuspended	1	The state of a machine learning task that,Äôs paused.

8.26 class MLUpdateContextMBS

8.26.1 class MLUpdateContextMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The context an update task provides to your app,Äôs completion and update progress handlers.
Blog Entries

- Update Machine Learning Model on Device
- MBS Xojo Plugins, version 19.6pr3

8.26.2 Methods

8.26.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.26.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The constructor.

8.26.5 writeToFile(file as FolderItem, byref error as NSErrorMBS) as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Exports a machine learning file to the file system.
Notes: file: The location in the file system where the file should be written.
Error: will return any error message.

Returns true on success or false on failure.
Requires MacOS 10.15 or newer.

8.26.6 Properties

8.26.7 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.26.8 Metrics as Dictionary

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The training metrics of the model for the update task, contained in a dictionary.
Notes: (Read only property)

8.26.9 Model as MLModelMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The underlying Core ML model stored in memory.
Notes: (Read only property)

8.26.10 Parameters as Dictionary

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The parameters for the update task.
Notes: (Read only property)

8.26.11 ProgressEvent as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The event type that triggered an update task to notify your app,Äôs completion and update progress handlers.
Notes: (Read only property)

8.26.12 Task as MLUpdateTaskMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The update task that generated the update context.
Notes: (Read only property)

8.26.13 Constants

Progress Event Types

Constant	Value	Description
ProgressEventEpochEnd	2	An event that represents the end of training epoch.
ProgressEventMiniBatchEnd	4	An event that represents the end of a mini-batch within a training epoch.
ProgressEventTrainingBegin	1	An event that represents the start of training.

8.27 class MLUpdateProgressHandlersMBS

8.27.1 class MLUpdateProgressHandlersMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A collection of closures an update task uses to notify your app of its progress.
Notes: Available in MacOS 10.15 or newer.
Blog Entries

- Update Machine Learning Model on Device
- MBS Xojo Plugins, version 19.6pr3

8.27.2 Methods

8.27.3 available as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Checks whether class is available.
Notes: Returns true in MacOS 10.15 or newer.

8.27.4 Constructor(interestedEvents as Integer)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates the collection of event handlers for an update task uses to notify your app of its progress.
Notes: interestedEvents: The events for which the update task will call your closures for, contained in an option set.

8.27.5 Properties

8.27.6 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

8.27.7 Events

8.27.8 completionHandler(context as MLUpdateContextMBS)

Plugin Version: 20.0, Platform: macOS, Targets: .

Function: The event that an update tasks uses to notify you when it is complete.

8.27.9 progressHandler(context as MLUpdateContextMBS)

Plugin Version: 20.0, Platform: macOS, Targets: .

Function: The event an update task uses to notify your app.
Notes: The update task only uses this event for the events you specified in interestedEvents.

8.27.10 Constants

Progress Event Types

Constant	Value	Description
ProgressEventEpochEnd	2	An event that represents the end of training epoch.
ProgressEventMiniBatchEnd	4	An event that represents the end of a mini-batch within a training epoch.
ProgressEventTrainingBegin	1	An event that represents the start of training.

8.28 class MLUpdateTaskMBS

8.28.1 class MLUpdateTaskMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: A task that updates a model with additional training data.
Notes: Use an MLUpdateTaskMBS to update a machine learning model on a user‚Äôs device.

Available on MacOS 10.15 or newer.
Subclass of the MLTaskMBS class.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.0
- Update Machine Learning Model on Device
- MBS Xojo Plugins, version 19.6pr3

8.28.2 Methods

8.28.3 Constructor(file as FolderItem, trainingData as MLArrayBatchProviderMBS, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates an update task for your model, given its file URL, training data, and your completion handler.
Notes: file: The location in the file system of a model file (ModelName.mlmodelc).
trainingData: The update data for the model, contained in a batch provider.
configuration: The model settings for a updated model object.

Calls later completionHandler event.
See also:

- 8.28.4 Constructor(file as FolderItem, trainingData as MLArrayBatchProviderMBS, configuration as MLModelConfigurationMBS, progressHandlers as MLUpdateProgressHandlersMBS, byref error as NSErrorMBS) 629

8.28.4 Constructor(file as FolderItem, trainingData as MLArrayBatchProviderMBS, configuration as MLModelConfigurationMBS, progressHandlers as MLUpdateProgressHandlersMBS, byref error as NSErrorMBS)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates an update task for your model, given its file URL, training data, and your progress handlers.
Notes: file: The location in the file system of a model file (ModelName.mlmodelc).
trainingData: The update data for the model, contained in a batch provider.
configuration: The model settings for a updated model object.
progressHandlers: The closures the task calls during the update process.
See also:

- 8.28.3 Constructor(file as FolderItem, trainingData as MLArrayBatchProviderMBS, configuration as MLModelConfigurationMBS, byref error as NSErrorMBS) 629

8.28.5 resumeWithParameters(updateParameters as Dictionary)

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Resumes a model update with updated parameter values.
Notes: updateParameters: Model training parameter values to replace those currently set in the update task.

Use this method to resume the model update task with newer parameter values. You use this method within the closures you provide in an MLUpdateProgressHandlersMBS instance to resume the MLUpdateTaskMBS.

8.28.6 Events

8.28.7 completionHandler(context as MLUpdateContextMBS)

Plugin Version: 20.0, Platform: macOS, Targets: .

Function: The complete event.
Notes: Called only if you use constructor without passing MLUpdateProgressHandlersMBS object, because if you use MLUpdateProgressHandlersMBS, the event is called there.

Chapter 9

HTMLViewer

9.1 class DesktopHTMLViewer

9.1.1 class DesktopHTMLViewer

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Class in Xojo for HTML rendering.
Notes: We have multiple methods for macOS, Windows and Linux, so please check each method.

9.1.2 Methods

9.1.3 WKWebViewMBS as WKWebViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Queries WKWebViewMBS associated to HTMLViewer.
Notes: Returns WKWebView for Xojo when used in HTMLViewer.

Chapter 10

HTMLViewer Linux

10.1 class HTMLViewer

10.1.1 class HTMLViewer

Plugin Version: 7.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Class in Xojo 2005 for HTML rendering.
Notes: We have multiple methods for macOS, Windows and Linux, so please check each method.

10.1.2 Methods

10.1.3 WKWebViewMBS as WKWebViewMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Queries WKWebViewMBS associated to HTMLViewer.
Notes: Returns WKWebView for Xojo when used in HTMLViewer.

Chapter 11

MediaLibrary

11.1 class MLMediaGroupMBS

11.1.1 class MLMediaGroupMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The MLMediaGroup class provides groupings for media objects from a single source of media, such as iTunes or Aperture.
Notes: The media objects—individual files containing a piece of media such as a photo, song, or movie—are referenced by one or more groups within each media source. These groupings serve as filters, providing hierarchical structure to the collection of objects in each source.

The structure of the group hierarchy is specific to each media source, but all sources have certain commonalities. For example, every source has a single root media group, which contains all groups and objects within that source. It is the highest-level parent group in the hierarchy and each of its descendant groups contains its own subgroups and their objects. All groups have a reference to their parent within the hierarchy. A group with no descendants contains only its own objects. If a media group does not contain any objects, it is not visible in the hierarchy.

A media group has an array of attributes which can change at any point. For example, a media group may have certain attributes that describe its objects, but these attributes appear only after the objects for that group have been loaded. When any media group attribute changes, observers are notified via KVO notification. For information about handling attributes that change, see Cocoa Bindings Programming Topics.

Every media group has a unique identifier as well as a type identifier. In certain cases, multiple groups within a source can have the same type identifier. For descriptions of group type identifiers, see MediaLibrary Constants.
All MLMediaGroup properties are read-only, so this information can be accessed but not altered.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- MBS Xojo Plugins, version 18.2pr1

11.1.2 Methods

11.1.3 childGroups as MLMediaGroupMBS()

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A list of child groups contained in the media group.

11.1.4 Constructor

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

11.1.5 mediaObjects as MLMediaObjectMBS()

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A list of media objects in the media group.
Notes: This accessor property is nonblocking. If there is no data yet, it returns nil and automatically triggers an internal asynchronous request. A KVO notification will be sent via the main thread when data arrives.

11.1.6 MLApertureAllPhotosTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents all photos in Aperture.

11.1.7 MLApertureAllProjectsTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents all projects in Aperture.

11.1.8 MLApertureFacebookAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Facebook album that is visible in Aperture.

11.1.9 MLApertureFacebookGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Facebook user account in Aperture. A Facebook user account contains one or more Facebook albums.

11.1.10 MLApertureFacesAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Faces album in Aperture. Individual Faces albums are nested in the main Faces album.

11.1.11 MLApertureFlaggedTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents the album of flagged media in Aperture.

11.1.12 MLApertureFlickrAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Flickr album that is visible in Aperture.

11.1.13 MLApertureFlickrGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Flickr user account in Aperture. A Flickr user account contains one or more Flickr albums.

11.1.14 MLApertureFolderAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a folder in Aperture.

11.1.15 MLApertureLastImportAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents the last import album in Aperture.

11.1.16 MLApertureLastNMonthsAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents the recent content album in Aperture, known as the Last N Months album. The value for N is usually 12 (settable in Aperture >Preferences >General).

11.1.17 MLApertureLastViewedEventAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents the last viewed event in Aperture.

11.1.18 MLApertureLightTableTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a light table in Aperture.

11.1.19 MLAperturePhotoStreamAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a photo stream in Aperture.

11.1.20 MLAperturePlacesAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents the Places album in Aperture.

11.1.21 MLAperturePlacesCityAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Places album for a city in Aperture. A city album is nested in a province or state album.

11.1.22 MLAperturePlacesCountryAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Places album for a country in Aperture. A country album is nested in the main Places album.

11.1.23 MLAperturePlacesPointOfInterestAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Places album for a point-of-interest in Aperture. A point of interest album is nested in a city album.

11.1.24 MLAperturePlacesProvinceAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a Places album for a province or state in Aperture. A province or state album is nested in a country album.

11.1.25 MLApertureProjectAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a project in Aperture.

11.1.26 MLApertureProjectFolderAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a folder within a project in Aperture.

11.1.27 MLApertureRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The root media group for Aperture.

11.1.28 MLApertureSlideShowTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: The media group that represents a slideshow in Aperture.

11.1.29 MLApertureSmugMugAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a SmugMug album that is visible in Aperture.

11.1.30 MLApertureSmugMugGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a SmugMug user account in Aperture. A SmugMug user account contains one or more SmugMug albums.

11.1.31 MLApertureUserAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a user-created album in Aperture.

11.1.32 MLApertureUserSmartAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Aperture media source.
Notes: A media group that represents a user-created smart album in Aperture.

11.1.33 MLFinalCutEventCalendarGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: A media group that represents a collection of events from a specific time period in Final Cut Pro.

11.1.34 MLFinalCutEventGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: A media group that represents an event in Final Cut Pro.

11.1.35 MLFinalCutEventLibraryGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: The media group that represents the event library in Final Cut Pro. The event library contains all event calendar groups.

11.1.36 MLFinalCutFolderGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: A media group that represents a folder in Final Cut Pro.

11.1.37 MLFinalCutProjectGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: A media group that represents a project in Final Cut Pro.

11.1.38 MLFinalCutRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Final Cut Pro media source.
Notes: The root media group for Final Cut Pro.

11.1.39 MLFolderGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in folder-based media sources.
Notes: A media group that represents a folder in folder-based media.

11.1.40 MLFolderRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in folder-based media sources.
Notes: The root media group for folder-based media.

11.1.41 MLGarageBandFolderGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the GarageBand media source.
Notes: A media group that represents a folder in GarageBand.

11.1.42 MLGarageBandRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the GarageBand media source.
Notes: The root media group for GarageBand.

11.1.43 MLiMovieEventCalendarGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: A media group that represents a collection of events from a specific time period in iMovie.

11.1.44 MLiMovieEventGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: A media group that represents an event in iMovie.

11.1.45 MLiMovieEventLibraryGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: The media group that represents the event library in iMovie. The event library contains all event calendar groups.

11.1.46 MLiMovieFolderGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: A media group that represents a folder in iMovie.

11.1.47 MLiMovieProjectGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: A media group that represents a project in iMovie.

11.1.48 MLiMovieRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iMovie media source.
Notes: The root media group for iMovie.

11.1.49 MLiPhotoAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents an album in iPhoto.

11.1.50 MLiPhotoEventAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents an event in iPhoto.

11.1.51 MLiPhotoEventsFolderTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the Events album in iPhoto.

11.1.52 MLiPhotoFacebookAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Facebook album that is visible in iPhoto.

11.1.53 MLiPhotoFacebookGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Facebook user account in iPhoto. A Facebook user account contains one or more Facebook albums.

11.1.54 MLiPhotoFacesAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Faces album in iPhoto. Individual Faces albums are nested in the main Faces album.

11.1.55 MLiPhotoFlaggedAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the album of flagged media in iPhoto.

11.1.56 MLiPhotoFlickrAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Flickr album that is visible in iPhoto.

11.1.57 MLiPhotoFlickrGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Flickr user account in iPhoto. A Flickr user account contains one or more Flickr albums.

11.1.58 MLiPhotoFolderAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a folder in iPhoto.

11.1.59 MLiPhotoLastImportAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the Last Import album in iPhoto.

11.1.60 MLiPhotoLastNMonthsAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the recent content album in iPhoto, known as the Last N Months album. The value for N is usually 12 (settable in iPhoto >Preferences >General).

11.1.61 MLiPhotoLastViewedEventAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the last viewed event in iPhoto.

11.1.62 MLiPhotoLibraryAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the Photos album in iPhoto.

11.1.63 MLiPhotoPhotoStreamAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a photo stream in iPhoto.

11.1.64 MLiPhotoPlacesAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The media group that represents the Places album in iPhoto.

11.1.65 MLiPhotoPlacesCityAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Places album for a city in iPhoto. A city album is nested in a province or state album.

11.1.66 MLiPhotoPlacesCountryAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Places album for a country in iPhoto. A country album is nested in the main Places album.

11.1.67 MLiPhotoPlacesPointOfInterestAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Places album for a point-of-interest in iPhoto. A point of interest

album is nested in a city album.

11.1.68 MLiPhotoPlacesProvinceAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a Places album for a province or state in iPhoto. A province or state album is nested in a country album.

11.1.69 MLiPhotoRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: The root media group for iPhoto.

11.1.70 MLiPhotoSlideShowAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a slideshow album in iPhoto.

11.1.71 MLiPhotoSmartAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a smart album in iPhoto.

11.1.72 MLiPhotoSubscribedAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iPhoto media source.
Notes: A media group that represents a subscribed album in iPhoto.

11.1.73 MLiTunesAudioBooksPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the Audio Books playlist in iTunes.

11.1.74 MLiTunesFolderPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: A media group that represents a folder in iTunes.

11.1.75 MLiTunesGeniusPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: A media group that represents a genius playlist in iTunes.

11.1.76 MLiTunesiTunesUPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the iTunes U playlist in iTunes.

11.1.77 MLiTunesMoviesPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the Movies playlist in iTunes.

11.1.78 MLiTunesMusicPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the Music playlist in iTunes.

11.1.79 MLiTunesMusicVideosPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: Musicvideos playlist

11.1.80 MLiTunesPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: A media group that represents a user-created playlist in iTunes.

11.1.81 MLiTunesPodcastPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the Podcast playlist in iTunes.

11.1.82 MLiTunesPurchasedPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the Purchased playlist in iTunes.

11.1.83 MLiTunesRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The root media group for iTunes.

11.1.84 MLiTunesSavedGeniusPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: A media group that represents a saved genius playlist in iTunes.

11.1.85 MLiTunesSmartPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: A media group that represents a smart playlist in iTunes.

11.1.86 MLiTunesTVShowsPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the iTunes media source.
Notes: The media group that represents the TV Shows playlist in iTunes.

11.1.87 MLiTunesVideoPlaylistTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: Video Playlist

11.1.88 MLLogicBouncesGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: The media group that represents all bounces in Logic.

11.1.89 MLLogicProjectsGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: The media group that represents all projects in Logic.

11.1.90 MLLogicProjectTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: A media group that represents a project in Logic. Projects may be nested.

11.1.91 MLLogicRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Logic media source.
Notes: The root media group for Logic.

11.1.92 MLPhotosAlbumsGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.93 MLPhotosAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.
Notes: A media group that represents an album in Photos.

11.1.94 MLPhotosAllCollectionsGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.95 MLPhotosAllMomentsGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.96 MLPhotosAllPhotosAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.97 MLPhotosAllYearsGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.98 MLPhotosAnimatedGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.99 MLPhotosBurstGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.100 MLPhotosCollectionGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.101 MLPhotosDepthEffectGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.102 MLPhotosFacesAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.103 MLPhotosFavoritesGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.104 MLPhotosFolderTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.105 MLPhotosFrontCameraGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.106 MLPhotosLastImportGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.107 MLPhotosLivePhotosGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.108 MLPhotosLongExposureGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.109 MLPhotosMomentGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.110 MLPhotosMyPhotoStreamTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.111 MLPhotosPanoramasGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.112 MLPhotosPublishedAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.113 MLPhotosRootGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.
Notes: The root media group for Photos.

11.1.114 MLPhotosScreenshotGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.115 MLPhotosSharedGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.116 MLPhotosSharedPhotoStreamTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.117 MLPhotosSloMoGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.118 MLPhotosSmartAlbumTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.
Notes: A media group that represents a smart album in Photos.

11.1.119 MLPhotosTimelapseGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.120 MLPhotosVideosGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.121 MLPhotosYearGroupTypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for media group types in the Photos media source.

11.1.122 Properties

11.1.123 Handle as Integer

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Internet object reference.
Notes: (Read and Write property)

11.1.124 IconImage as NSImageMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The media group,Äôs icon.
Notes: (Read only property)

11.1.125 Identifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An identifier for the media group.
Notes: Each group,Äôs identifier is unique within a media source.
(Read only property)

11.1.126 MediaLibrary as MLMediaLibraryMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A pointer to the media library instance that loaded the media group,Äôs source.
Notes: (Read only property)

11.1.127 MediaSourceIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An identifier for the source that loaded the media group.
Notes: (Read only property)

11.1.128 ModificationDate as Date

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time when the media group was last altered.
Notes: (Read only property)

11.1.129 ModificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time when the media group was last altered.
Notes: (Read only property)

11.1.130 Name as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The name of the media group.
Notes: This string is human-readable. It is either user created (such as the name of an iTunes playlist) or already localized.
(Read only property)

11.1.131 Parent as MLMediaGroupMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The media group,Äôs parent group.
Notes: (Read only property)

11.1.132 Properties as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A dictionary of attributes describing the media group.
Notes: These attributes are usually defined by the source app, such as iTunes. For example, an iTunes playlist is represented as a group. iTunes attaches attributes such as "Playlist Persistent ID" to the group in its attributes. The attribute names vary based on the media source. Attributes common to all sources are called out as separate properties.
(Read only property)

11.1.133 TypeIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An identifier for the media group,Äôs type.
Notes: Multiple groups within a media source can have the same type identifier. For descriptions of group type identifiers, see MediaLibrary Constants.
(Read only property)

11.1.134 URL as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the media group.
Notes: Some groups do not have a URL, in which case this returns nil. For example, a group that represents a filesystem folder on disk has a URL, but a group that represents a named face in iPhoto does not.
(Read only property)

11.2 class MLMediaLibraryMBS

11.2.1 class MLMediaLibraryMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The MLMediaLibrary class provides an interface for accessing a collection of media objects from various sources.
Notes: It serves as the initial access point of the Media Library framework.

The media library structure is defined by MLMediaSource, MLMediaGroup, and MLMediaObject classes. At the highest level, all content within a media library instance is categorized by media source. Conceptually, a media source represents a single app, such as iTunes or Aperture. Each source contains a hierarchy of media groups that originates from a root group. These groups consist of media objects—individual files containing a piece of media such as a photo, song, or movie. Only one copy of each object exists within a media library instance, but an object can be referenced by multiple groups from a single source. The structure of the group hierarchy is specific to each media source.

A media library is initialized using the Constructor. The options argument to this method serves as a filter. By specifying which folders or sources to include or exclude during load, you can view a particular subset of groups and objects from your collection. All objects provided are thread-safe. For descriptions of possible load options, see Load Options Keys.
The typical and most efficient use case is to create and use one instance of MLMediaLibrary for the lifetime of an app. When the underlying media files and metadata on the user,Äôs system change, the corresponding data model objects (media groups and media objects) are automatically updated and KVO notifications are sent to notify the calling code of any changes. Multiple instances of MLMediaLibrary can be created and used, but their sources, groups, and objects will be independent of those provided by other instances of MLMediaLibrary.
Blog Entries

- MBS Xojo Plugins, version 22.5pr2
- MBS Xojo Plugins, version 18.2pr1

11.2.2 Methods

11.2.3 Available as Boolean

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether this class available.
Notes: Returns true on Mac OS 10.7 or newer in 64-bit app.

11.2.4 Constructor(options as Dictionary)

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the media library based on the specified load options.
Notes: options: A dictionary of load options.

Returns a new media library.

For descriptions of possible load options, see Load Options Keys.

11.2.5 MLMediaLoadAppFoldersKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the option keys.
Notes: Specifies one or more relative paths inside the caller‚Äôs app bundle in which to search for media files. The value for this key is an array of strings (relative paths inside the caller‚Äôs app bundle).

11.2.6 MLMediaLoadAppleLoops as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for well-known media folders used to specify the value for MLMediaLoadFoldersKey.
Notes: Identifies the folder containing audio loops from Apple.

11.2.7 MLMediaLoadExcludeSourcesKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the option keys.
Notes: Defines which media sources to exclude when loading. This option is processed after MLMediaLoadIncludeSourcesKey. The value for this key is an array of strings (media source identifiers). For a list of valid media source identifiers, see Media Source Identifiers.

11.2.8 MLMediaLoadFoldersKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the option keys.
Notes: Specifies the well-known folders that should be searched for media files. If this key is not present, none of the well-known folders will be provided. The value for this key is an array of strings (identifiers that correspond to well-known folder locations). For a list of well-known folder identifiers, see MLMediaLoadMoviesFolder and MLMediaLoadAppleLoops.

11.2.9 MLMediaLoadIncludeSourcesKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the option keys.
Notes: Defines which media sources to include when loading. If not present, load all available media sources. This option is processed after MLMediaLoadSourceTypesKey. If MLMediaLoadIncludeSourcesKey is present but MLMediaLoadSourceTypesKey is not, then only those sources specified here will be loaded. This is useful for loading a single media source. When both keys are present, this is useful for adding one or more media sources that normally would not appear for the requested library type. The value for this key is an array of strings (media source identifiers). For a list of valid media source identifiers, see Media Source Identifiers.

11.2.10 MLMediaLoadMoviesFolder as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the identifiers for well-known media folders used to specify the value for MLMediaLoadFoldersKey.
Notes: Identifies the user,Äôs Movies folder.

11.2.11 MLMediaLoadSourceTypesKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the option keys.
Notes: Defines which sources to load based on library type. If not present, this will load all sources. The value for this key is a media source type.

11.2.12 Properties

11.2.13 Handle as Integer

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Internet object reference.
Notes: (Read and Write property)

11.2.14 mediaSources as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a dictionary of media sources by identifier.
Notes: Returns nil the first time, beginning an asynchronous load of the media sources. A KVO notification is sent when all media sources have been loaded. If there are no objects in a media source, the source does not appear in this dictionary.
(Read only property)

11.3 class MLMediaObjectMBS

11.3.1 class MLMediaObjectMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The MLMediaObject class describes a single media file, such as a photo, song, or movie.
Notes: Each media object contains basic metadata including a name, media type, URL, and so on. Additional information about each object is stored in its list of attributes. For a list of possible object attribute keys, see Media Object Attribute Keys.

A media object belongs to a single media source but can be referenced by several groups within that source. In other words, an object can appear in multiple places in the group hierarchy under a single media source. In iTunes, a movie that was purchased through the iTunes Store is referenced by both the Purchased playlist and the Movies playlist. If a user adds the movie to his own playlist, the group respresenting that playlist will also reference the movie. All three groups reference the same media object.

All MLMediaObject properties are read-only, so this information can be accessed but not altered.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 18.2pr1

11.3.2 Methods

11.3.3 Constructor

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

11.3.4 MLMediaObjectAlbumKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs album. The value for this key is a string.

11.3.5 MLMediaObjectArtistKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs artist. The value for this key is a string.

11.3.6 MLMediaObjectBitRateKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs bit rate, in kilobits per second. The value for this key is a number.

11.3.7 MLMediaObjectChannelCountKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs channel count. The value for this key is a number.

11.3.8 MLMediaObjectCommentsKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the contents of the comments field associated with the media object. The value for this key is a string.

11.3.9 MLMediaObjectDurationKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs duration, in seconds. The value for this key is a number.

11.3.10 MLMediaObjectGenreKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs genre. The value for this key is a string.

11.3.11 MLMediaObjectKeywordsKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the keywords associated with the media object. The value for this key is an array of strings.

11.3.12 MLMediaObjectKindKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Used by iTunes only. Specifies the media object,Äôs file format (shown in the ,ÄúKind,Äù column in iTunes). The value for this key is a string.

11.3.13 MLMediaObjectProtectedKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies whether the media object is protected by DRM (Digital Rights Management). The value for this key is a boolean value.

11.3.14 MLMediaObjectResolutionStringKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs resolution. The value for this key is a string with size.

11.3.15 MLMediaObjectSampleRateKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs sample rate, in samples per second (Hz). The value for this key is a number.

11.3.16 MLMediaObjectTrackNumberKey as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys for a media object.
Notes: Specifies the media object,Äôs track number. The value for this key is a number.

11.3.17 Properties

11.3.18 ArtworkImage as NSImageMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Album artwork associated with the media object.
Notes: Applies to iTunes media only. Returns nil if not applicable or not available.
(Read only property)

11.3.19 ContentType as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The UTI associated with the media object.
Notes: (Read only property)

11.3.20 File as FolderItem

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the media object.
Notes: For your convenience, the plugin provides here a folderitem for the URL.
(Read only property)

11.3.21 FileSize as UInt64

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The size, in bytes, of the media object.
Notes: (Read only property)

11.3.22 Handle as Integer

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Internet object reference.
Notes: (Read and Write property)

11.3.23 Identifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An identifier for the media object.
Notes: Each object,Äôs identifier is unique within a media source.
(Read only property)

11.3.24 MediaLibrary as MLMediaLibraryMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A pointer to the media library instance that loaded the media object,Äôs source.
Notes: (Read only property)

11.3.25 MediaSourceIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An identifier for the source that loaded the media object.
Notes: For a list of possible media source identifiers, see MLMediaLibraryMBS.
(Read only property)

11.3.26 MediaType as Integer

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The media object,Äôs type of media (image, audio, or movie).
Notes: For a list of possible media types, see kType* constants.
(Read only property)

11.3.27 ModificationDate as Date

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time when the media object was last altered.
Notes: (Read only property)

11.3.28 ModificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time when the media object was last altered.
Notes: (Read only property)

11.3.29 Name as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The name of the media object.
Notes: (Read only property)

11.3.30 OriginalFile as FolderItem

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the original media object, if URL is not the original (master).
Notes: For your convenience, the plugin provides here a folderitem for the URL.
(Read only property)

11.3.31 OriginalURL as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the original media object, if URL is not the original (master).
Notes: This property is provided as a security-scoped URL. In order to gain access to the file that this URL refers to, the caller must call startAccessingSecurityScopedResource before and stopAccessingSecurityScopedResource after using the URL to access the file. For more information about security-scoped URLs, see NSURL.
(Read only property)

11.3.32 Properties as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A dictionary of attributes describing the media object.
Notes: For a list of possible object attribute keys, see Media Object Attribute Keys.
(Read only property)

11.3.33 ThumbnailFile as FolderItem

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the media object,Äôs thumbnail image.
Notes: For your convenience, the plugin provides here a folderitem for the URL.
(Read only property)

11.3.34 ThumbnailURL as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the media object,Äôs thumbnail image.
Notes: This property is provided as a security-scoped URL. In order to gain access to the file that this URL refers to, the caller must call startAccessingSecurityScopedResource before and stopAccessingSecurityScopedResource after using the URL to access the file. For more information about security-scoped URLs, see NSURL.
(Read only property)

11.3.35 URL as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The location of the media object.
Notes: This property is provided as a security-scoped URL. In order to gain access to the file that this URL refers to, the caller must call startAccessingSecurityScopedResource before and stopAccessingSecurityScopedResource after using the URL to access the file. For more information about security-scoped URLs, see NSURL.
(Read only property)

11.3.36 Constants

Types

Constant	Value	Description
kTypeAudio	1	Audio
kTypeImage	2	Image
kTypeMovie	4	Movie

11.4 class MLMediaSourceMBS

11.4.1 class MLMediaSourceMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The MLMediaSource class identifies a specific provider of media.
Notes: Conceptually, a media source respresents a single app, such as iTunes or Aperture. Each media source contains multiple groups of media objects—individual files containing a piece of media such as a photo, song, or movie.

The structure of the group hierarchy is specific to each media source, but all sources have certain commonalities. For example, every source has a single root media group, which contains all groups and objects within that source. It is the highest-level parent group in the hierarchy and each of its descendant groups contains its own subgroups and their objects. All groups have a reference to their parent within the hierarchy. A group with no descendants contains only its own objects. If a media group does not contain any objects, it is not visible in the hierarchy.
Every media source has a unique media source identifier within a single media library instance. For a list of possible media source identifiers, see Media Source Identifiers.
All MLMediaSourceMBS properties are read-only, so this information can be accessed but not altered.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 18.2pr1

11.4.2 Methods

11.4.3 Constructor

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

11.4.4 mediaGroupForIdentifier(mediaGroupIdentifier as string) as MLMediaGroupMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the media group with the specified identifier.
Notes: mediaGroupIdentifier: The media group identifier to search for in the source.

The media source must have finished loading before this method returns valid data. Specifically, the root media group must be available before the lookup methods will succeed. Otherwise, the return value is undefined.

11.4.5 mediaGroupsForIdentifiers(mediaGroupIdentifiers() as string) as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the media groups with the specified identifiers.
Notes: mediaGroupIdentifiers: An array of media group identifiers to search for in the source.

Returns a dictionary of media groups matching the specified identifiers.

The media source must have finished loading before this method returns valid data. Specifically, the root media group must be available before the lookup methods will succeed. Otherwise, the return value is undefined.

11.4.6 mediaObjectForIdentifier(mediaObjectIdentifier as string) as MLMediaObjectMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the media object with the specified identifier.
Notes: mediaObjectIdentifier: The media object identifier to search for in the media source.

The media source must have finished loading before this method returns valid data. Specifically, the root media group must be available before the lookup methods will succeed. Otherwise, the return value is undefined.

11.4.7 mediaObjectsForIdentifiers(mediaObjectIdentifiers() as string) as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the media objects with the specified identifiers.
Notes: mediaObjectIdentifiers: An array of media object identifiers to search for in the source.

Returns a dictionary of media objects matching the specified identifiers.

The media source must have finished loading before this method returns valid data. Specifically, the root media group must be available before the lookup methods will succeed. Otherwise, the return value is undefined.

11.4.8 MLMediaSourceApertureIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from Aperture.

11.4.9 MLMediaSourceAppDefinedFoldersIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from app defined folder.

11.4.10 MLMediaSourceCustomFoldersIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from a custom folder.

11.4.11 MLMediaSourceFinalCutIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from Final Cut Pro.

11.4.12 MLMediaSourceGarageBandIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from GarageBand.

11.4.13 MLMediaSourceiMovieIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from iMovie.

11.4.14 MLMediaSourceiPhotoIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from iPhoto.

11.4.15 MLMediaSourceiTunesIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from iTunes.

11.4.16 MLMediaSourceLogicIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from Logic.

11.4.17 MLMediaSourceMoviesFolderIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from movies folder.

11.4.18 MLMediaSourcePhotoBoothIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from Photo Booth.

11.4.19 MLMediaSourcePhotosIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The identifier for the media source providing content from photos folder.

11.4.20 Properties

11.4.21 Handle as Integer

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Internet object reference.
Notes: (Read and Write property)

11.4.22 MediaLibrary as MLMediaLibraryMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A pointer to the media library instance that loaded this media source.
Notes: (Read only property)

11.4.23 mediaSourceIdentifier as String

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A unique identifier for the media source.
Notes: For a list of possible media source identifiers, see shared methods.
(Read only property)

11.4.24 Properties as Dictionary

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A list of attributes describing the media source.
Notes: (Read only property)

11.4.25 rootMediaGroup as MLMediaGroupMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The base media group in the media source that contains all other groups within the source as descendant elements.
Notes: This accessor property is nonblocking. If there is no data yet, it returns nil and automatically triggers an internal asynchronous request. When data arrives, a KVO notification is sent via the main thread.
(Read only property)

11.4.26 Constants

Source Types

Constant	Value	Description
kSourceTypeAudio	1	Audio
kSourceTypeImage	2	Images
kSourceTypeMovie	4	Movies

Chapter 12

Photos

12.1 control DesktopPHLivePhotoControlMBS

12.1.1 control DesktopPHLivePhotoControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A view that displays a Live Photo—a picture that also includes motion and sound from the moments just before and after its capture.
Notes: Use a Live Photo view to display the photo and control playback of its motion and sound content.

In macOS, Live Photo objects are available only when editing Live Photo content in a photo editing extension that runs in the Photos app—see the PHContentEditingInputMBS class to access Live Photo content in an editing session.

By default, a Live Photo view uses its own gesture recognizer to allow the user to play the motion and sound content of a Live Photo with the same interactions and visual effects seen in the Photos app. To customize this gesture recognizer—for example, to install it on a different view for proper event handling in your app,Äôs view hierarchy—use the playbackGestureRecognizer property.

To animate the view briefly to hint that a picture is a Live Photo, use the startPlayback method with the PlaybackStyleHint option.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- New in the MBS Xojo Plugins Version 20.2
- MBS Xojo Plugins, version 20.2pr1

Xojo Developer Magazine

- 18.3, page 57: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

12.1.2 Methods

12.1.3 available as boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

12.1.4 startPlayback(style as integer)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Begins playback of Live Photo content in the view.
Notes: style: An option for how much of the Live Photo,Äôs motion and sound content to play. See PHLivePhotoViewPlaybackStyle.

Use the style parameter to choose whether to play the full motion and sound content of the Live Photo or only a brief section.

Typically, an app does not need to directly control playback, because a Live Photo view provides interactive playback control. Use this method only when non-interactive playback is appropriate—for example, to briefly animate the content to indicate that a view contains a Live Photo rather than a still image.

12.1.5 stopPlayback

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Ends playback of Live Photo content in the view.
See also:

- 12.1.6 stopPlayback(animated as boolean) 680

12.1.6 stopPlayback(animated as boolean)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Ends playback of Live Photo content in the view.
See also:

- 12.1.5 stopPlayback 680

12.1.7 Properties

12.1.8 audioVolume as Single

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The audio gain to apply to the Live Photo,Äôs movie content during playback.
Notes: Values for this property must be between 0.0 and 1.0, inclusive. A value of 1.0 (the default) plays audio content from the Live Photo at full volume (relative to the system volume). A value of 0.0 is equivalent to setting the muted property to true.
(Read and Write property)

12.1.9 ContentMode as Integer

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The content mode.
Notes: (Read and Write property)

12.1.10 LivePhoto as PHLivePhotoMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The Live Photo displayed in the view.
Notes: (Read and Write property)

12.1.11 livePhotoBadgeView as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A view for displaying Live Photo status.
Notes: The PHLivePhotoViewMBS uses this subview to display icons indicating the existence or status of Live Photo content. Photos manages the content displayed in the badge view, so you don,Äôt need to do anything with view,Äôs content—instead, this property provides access to the badge view so you can change where it appears in your view hierarchy if needed. For example, if you display a Live Photo view within a scroll view, you can move the badge view so that its position remains constant while the scroll view scrolls.

(Read only property)

12.1.12 muted as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines whether the view plays the audio content of its Live Photo.
Notes: The default value is false, indicating that the view plays audio content along with the motion content of its Live Photo. Change this value to true to play motion content but not audio content.
(Read and Write property)

12.1.13 View as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.
Notes: Use this object to set more options on the control.
(Read only property)

12.1.14 Events

12.1.15 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

12.1.16 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

12.1.17 didEndPlaybackWithStyle(playbackStyle as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Notifies the delegate that Live Photo playback has ended.
Notes: playbackStyle: The style of playback, indicating whether the content was played in full or briefly previewed.

12.1.18 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.
In older Xojo versions, this event is named LostFocus.
Notes:

This only fires if the control itself lost focus and not a sub control.

12.1.19 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.
In older Xojo versions, this event is named GotFocus.
Notes:

This only fires if the control itself got focus and not a sub control.

12.1.20 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

12.1.21 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named EnableMenuItems.

12.1.22 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

12.1.23 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

12.1.24 MouseUp(x As Integer, y As Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

12.1.25 ScaleFactorChanged(NewFactor as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

12.1.26 willBeginPlaybackWithStyle(playbackStyle as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Notifies the control that Live Photo playback is about to begin.
Notes: playbackStyle: The style of playback, indicating whether the content is to be played in full or briefly previewed.

12.1.27 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

12.1.28 Constants

Content Modes

Constant	Value	Description
ContentModeAspectFill	1	Aspect Fill.
ContentModeAspectFit	0	Aspect Fit.

Playback Styles

Constant	Value	Description
PlaybackStyleFull	1	Plays back the entire motion and sound content of the Live Photo, including transition effects at the start and end.
PlaybackStyleHint	2	Plays back only a brief section of the motion content of the Live Photo, without sound.
PlaybackStyleUndefined	0	This value is invalid for use.

12.2 class PHAdjustmentDataMBS

12.2.1 class PHAdjustmentDataMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A description of the edits made to an asset's photo, video, or Live Photo content, which allows your app to reconstruct or revert the effects of prior editing sessions.
Notes: When a user edits an asset, Photos saves a PHAdjustmentData object along with the modified image or video data. This object provides an application-defined ,Äúrecipe,Äù you can use to reconstruct the edit. For example, if your app applies filters to a photo, you might might create adjustment data that identifies which filters the user picked, the parameters for each, and the order to apply the filters in. Later, the user can resume working with those filters and parameters by using your app or another app that understands your adjustment data format. When iCloud Photos is enabled, a user can revert or resume edits made on a different device.

You work with adjustment data when editing an asset, using either the requestContentEditingInputWithOptions method or a photo extension view controller that implements the PHContentEditingController protocol.

- When you begin an edit (through a PHContentEditingInputMBS object), examine the editing input,Äôs adjustmentData property to decide whether the last edit made to the asset is compatible with your app. If so, you can allow the user to resume working with that edit. If not, you can make further edits to the last saved version of the photo.

- When you commit an edit (through a PHContentEditingOutputMBS object), provide a new adjustment whose data represents the changes your app made.

For each asset, Photos stores only one PHAdjustmentDataMBS object, representing the most recent edit made to the asset,Äôs content.

12.2.2 Methods

12.2.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.11 or newer.

12.2.4 Constructor(formatIdentifier as string, formatVersion as string, data as Memoryblock)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes an adjustment object with the specified format and data.
Notes: formatIdentifier: A string that uniquely identifies the format of the adjustment data.
formatVersion: A version number for the adjustment data format.
data: A serialized form of whatever information is needed to reconstruct the adjustment.

Returns an initialized adjustment object.

To uniquely identify the adjustments your app makes, use the the formatIdentifier and formatVersion parameters. These parameters help you determine whether and how to interpret the adjustment data when working with an edited asset later. For best results, use reverse-DNS-style identifiers and monotonically increasing version numbers.
For example, in the first version of your app, you might save adjustment data using the identifier com.example.myApp and version 1.0. If a later version of your app adds incompatible information to the adjustment data, you can use the same identifier and increase the version number to 2.0.

Use the data parameter to store whatever information is useful to your app for reconstructing an edit. For example, if your app applies Core Image filters to photos, you can use this parameter to store a serialized property list that describes the filters and their parameters.

Because Photos limits the size of adjustment data, you should keep your edit information short and descriptive. Don,Äôt use image data to describe an edit—instead, save only the minimal information that is needed to recreate the edit.
Your app must provide a non-empty Memoryblock for the data parameter. If you cannot provide relevant data to describe an edit, you may pass data that encodes an NSUUID object.

12.2.5 Properties

12.2.6 data as Memoryblock

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Data that contains the information necessary to reconstruct the adjustment.
Notes: Use this property to resume working with the last edit that was made to an asset. For example, if your app applies Core Image filters to photos, this property may hold a serialized property list that describes the filters and their parameters. Use the formatIdentifier and formatVersion properties to determine whether the adjustment data saved with an asset is in a format that your app can understand.
(Read only property)

12.2.7 formatIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A string uniquely identifying the format of the adjustment data.
Notes: Set this identifier when you create an adjustment object with the constructor. For best results, identify your organization or product using a reverse-DNS-style name, such as com.example.myApp.

Read this property, and the formatVersion property, to determine whether the adjustment data saved with an asset was created by your app or is otherwise compatible with your app.
(Read only property)

12.2.8 formatVersion as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A version number for the adjustment data format.
Notes: Set this identifier when creating an adjustment object with the constructor.
Read this property, and the formatIdentifier property, to determine whether the adjustment data saved with an asset was created by your app or is otherwise compatible with your app.
For example, in the first version of your app, you might save adjustment data using the identifier com.example.myApp and version 1.0. If a later version of your app adds incompatible information to the adjustment data, you can use the same identifier and increase the version number to 2.0.
(Read only property)

12.2.9 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.3 class PHAssetChangeRequestMBS

12.3.1 class PHAssetChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A request to create, delete, change metadata for, or edit the content of a Photos asset, for use in a photo library change block.
Notes: You use the PHAssetChangeRequest class to request changes for PHAssetMBS objects. To make changes to assets in the Photos library, create a change request by using the appropriate class method for the change you want to perform.

- Call one of the methods listed in Adding New Assets to create a new asset from an image or video file.
- Call the deleteAssets: method to delete existing assets.
- Call the changeRequestForAsset: method to modify an asset,Äôs content or metadata.

A change request for creating or modifying an asset works like a mutable version of the asset object. Use the change request,Äôs properties to request changes to the corresponding properties of the asset itself.

After Photos runs the change block and calls your completion handler, the asset,Äôs state reflects the changes that you requested in the block.

If you create or use a change request object outside a photo library change block, Photos raises an Objective-C exception. For details on change blocks, see PHPhotoLibraryMBS.

12.3.2 Methods

12.3.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.3.4 changeRequestForAsset(asset as PHAssetMBS) as PHAssetChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for modifying the specified asset.
Notes: asset: The asset to be modified.

Returns an asset change request.

Before editing an asset, use its canPerformEditOperation method to see if the asset allows editing.

After you create a change request within a photo library change block, propose changes to the original asset,Äôs properties by setting the corresponding properties of the change request. After Photos runs your change block, the asset,Äôs properties reflect your changes. For details on change blocks, see PHPhotoLibraryMBS.

To edit an asset,Äôs image or video content, first begin a content editing session with the asset,Äôs requestContentEditingInputWithOptions method. You commit a content edit by setting the contentEditingOutput property of a change request within a change block. For more information about asset content editing, see PHAssetMBS.

12.3.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.3.6 creationRequestForAssetFromImage(image as NSImageMBS) as PHAssetChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new image asset to the Photos library.
Notes: image: An image.

Returns an asset creation request.

Call this method within a photo library change block to create a new asset. For details on change blocks, see PHPhotoLibraryMBS.
To reference the newly created asset later in the same change block or after the change block completes, use the placeholderForCreatedAsset property to retrieve a placeholder object.

12.3.7 creationRequestForAssetFromImageAtFile(file as FolderItem) as PHAssetChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new image asset to the Photos library, using the image file at the specified URL.
Notes: file: A URL for an image file.

Returns an asset creation request.

Call this method within a photo library change block to create a new asset. For details on change blocks, see PHPhotoLibraryMBS.

To set metadata properties of the newly created asset, use the corresponding properties of the change request (listed in Modifying Assets). To reference the newly created asset later in the same change block or after the change block completes, use the placeholderForCreatedAsset property to retrieve a placeholder object.

12.3.8 creationRequestForAssetFromVideoAtFileURL(file as FolderItem) as PHAssetChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new video asset to the Photos library, using the video file at the specified URL.
Notes: file: A folderitem for a video file.

Returns an asset creation request.

Call this method within a photo library change block to create a new asset. For details on change blocks, see PHPhotoLibraryMBS.

To set metadata properties of the newly created asset, use the corresponding properties of the change request (listed in Modifying Assets). To reference the newly created asset later in the same change block or after the change block completes, use the placeholderForCreatedAsset property to retrieve a placeholder object.

12.3.9 deleteAssets(assets() as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests that the specified assets be deleted.
Notes: assets: An array of PHAssetMBS objects to be deleted.

Call this method within a photo library change block to delete assets. For details on change blocks, see PHPhotoLibraryMBS.

12.3.10 revertAssetContentToOriginal

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Request to revert any edits made to the asset,Äôs content.
Notes: When an asset has been edited, Photos stores multiple versions of the asset: the original version of the asset as it was first captured or imported, and the input and output of the most recent edit. (You work with asset versions when requesting to edit an asset,Äôs content—see PHContentEditingInputRequestOptions.) Call this method to revert to the original version of the asset, discarding all edits.

This request fails if original content for the asset is not available on the current device (for example, if iCloud Photo Library is enabled and the user has edited the asset on a different device). Use PHAssetResourceManagerMBS to ensure that original asset content is downloaded to the current device before making this request.

12.3.11 Properties

12.3.12 contentEditingOutput as PHContentEditingOutputMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The output of an asset content editing session.
Notes: To edit an asset,Äôs image or video content, you must first begin a content editing session with the asset,Äôs requestContentEditingInputWithOptions method. You commit a content edit by setting the contentEditingOutput property of a change request within a change block. For more information about asset content editing, see PHAssetMBS.
(Read and Write property)

12.3.13 creationDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The date and time at which the asset claims to have been originally created.
Notes: Set this property to change the creation date of an asset.
(Read and Write property)

12.3.14 creationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The date and time at which the asset claims to have been originally created.
Notes: Set this property to change the creation date of an asset.
(Read and Write property)

12.3.15 Favorite as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the asset is marked as one of the user,Äôs favorites.
Notes: (Read and Write property)

12.3.16 Hidden as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the asset is hidden in collections.
Notes: Hidden assets do not appear in the Moments interface in the Photos app.
(Read and Write property)

12.3.17 location as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The location information saved with the asset.
Notes: Typically, an asset,Äôs location metadata identifies the place where the asset was captured. Use this property to provide a different location.

We pass result, the CLLocationMBS object, as variant to avoid plugin dependencies.
(Read and Write property)

12.3.18 placeholderForCreatedAsset as PHObjectPlaceholderMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A placeholder object for the asset that the change request creates.
Notes: Use this property if you need to reference the asset created by a change request within the same change block.
(Read only property)

12.4 class PHAssetCollectionChangeRequestMBS

12.4.1 class PHAssetCollectionChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A request to create, delete, or modify a Photos asset collection, for use in a photo library change block.
Notes: You use the PHAssetCollectionChangeRequestMBS class to request changes for PHAssetCollection-MBS objects. To make changes to asset collections (such as user-created albums) in the Photos library, create a change request using the appropriate class method for the change you want to perform.

- Call the creationRequestForAssetCollectionWithTitle: method to create a new asset collection.
- Call the deleteAssetCollections: method to delete existing asset collections.
- Call the changeRequestForAssetCollection: or changeRequestForAssetCollection:assets: method to modify a collection,Äôs metadata or list of member assets.

Before creating a change request, use the canPerformEditOperation: method to verify that the collection allows the edit operation you,Äôre requesting. If you attempt to perform an unsupported edit operation, Photos throws an exception.
A change request for creating or modifying an asset collection works like a mutable version of the asset collection object. Use the change request,Äôs properties and instance methods to request changes to the asset collection itself.

After Photos runs the change block and calls your completion handler, the asset collection,Äôs state reflects the changes you requested in the block.
If you create or use a change request object outside a photo library change block, Photos raises an Objective-C exception. For details on change blocks, see PHPhotoLibraryMBS.
Subclass of the PHChangeRequestMBS class.

12.4.2 Methods

12.4.3 addAsset(asset as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Add the specified asset to the asset collection.
Notes: asset: The PHAsset object to be added to the asset collection.

If you created the change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method, Photos inserts the new assets after the existing assets in the collection.

Otherwise, the arrangement of the new assets relative to others in the collection is undefined.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.4 addAssets(assets() as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Add the specified assets to the asset collection.
Notes: assets: An array of PHAsset objects to be added to the asset collection.

If you created the change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method, Photos inserts the new assets after the existing assets in the collection. Otherwise, the arrangement of the new assets relative to others in the collection is undefined.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.5 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.4.6 changeRequestForAssetCollection(assetCollection as PHAssetCollectionMBS) as PHAssetCollectionChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for modifying the specified asset collection.
Notes: assetCollection: The asset collection to be modified.

Returns an asset collection change request.

After you create a change request within a photo library change block, propose changes to the collection,Äôs title or list of member assets with the properties and instance methods of the change request. After Photos

runs your change block, the asset collection reflects your changes. For details on change blocks, see PHPhotoLibraryMBS.

Use this method when modifying an asset collection,Äôs metadata or when adding or removing assets without regard to their arrangement. To work with indexes in the list of member assets, use the changeRequestForAssetCollection method instead.
See also:

- 12.4.7 changeRequestForAssetCollection(assetCollection as PHAssetCollectionMBS, assets as PHFetchResultMBS) as PHAssetCollectionChangeRequestMBS 696

12.4.7 changeRequestForAssetCollection(assetCollection as PHAssetCollectionMBS, assets as PHFetchResultMBS) as PHAssetCollectionChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for modifying the specified asset collection, with a fetch result for tracking changes.
Notes: assetCollection: The asset collection to be modified.
assets: A fetch result listing the assets in the collection.

Returns an asset collection change request.

After you create a change request within a photo library change block, you propose changes to the collection,Äôs title or list of member assets with the properties and instance methods of the change request. After Photos runs your change block, the asset collection reflects your changes. For details on change blocks, see PHPhotoLibraryMBS.

Use this method when you need to insert, remove, or rearrange assets at specified indexes in the asset collection,Äôs list of member assets. By passing in a fetch result reflecting what your app sees as the current state of the collection,Äôs membership, the Photos framework can ensure that the indexes you specify are valid even if the collection has changed since you last fetched it. If you don,Äôt need to work with indexes in the list of member assets, you can use the changeRequestForAssetCollection method instead.
See also:

- 12.4.6 changeRequestForAssetCollection(assetCollection as PHAssetCollectionMBS) as PHAssetCollectionChangeRequestMBS 695

12.4.8 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.4.9 creationRequestForAssetCollectionWithTitle(title as string) as PHAssetCollectionChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new asset collection to the Photos library.
Notes: image: A name for the new asset collection.

Returns an asset collection creation request.

Call this method within a photo library change block to create a new asset collection. For details on change blocks, see PHPhotoLibraryMBS.
To add assets to the newly created asset collection or change its title, use the methods listed in Modifying Asset Collections. To reference the newly created asset collection later in the same change block or after the change block completes, use the placeholderForCreatedAssetCollection property to retrieve a placeholder object.

12.4.10 deleteAssetCollections(Collections() as PHCollectionListMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests that the specified asset collections be deleted.
Notes: assetCollections: An array of PHAssetCollectionMBS objects to be deleted.

Call this method within a photo library change block to delete asset collections. For details on change blocks, see PHPhotoLibraryMBS.

12.4.11 insertAsset(asset as PHAssetMBS, index as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Insert the specified asset into the collection at the specified indexes.
Notes: asset: A PHAssetMBS object to be inserted into the asset collection.
index: The index at which the asset should be inserted. The count of locations in this index set must equal the count of assets.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before inserting assets.

For a detailed discussion of how the index set you specify maps to insertions in the list of assets, see the similar NSMutableArray method insertObjects.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.12 insertAssets(assets() as PHAssetMBS, indexes as NSIndexSetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Inserts the specified assets into the collection at the specified indexes.
Notes: assets: An array of PHAssetMBS objects to be inserted into the asset collection.
indexes: The indexes at which the assets should be inserted. The count of locations in this index set must equal the count of assets.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before inserting assets.

For a detailed discussion of how the index set you specify maps to insertions in the list of assets, see the similar NSMutableArray method insertObjects.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.13 moveAsset(fromIndex as Integer, toIndex as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Move the asset at the specified index in the asset collection to a new index.
Notes: fromIndex: The indexes of the asset to be moved in the asset collection.
toIndex: The index at which to place the moved asset, relative to the collection,Äôs ordering after removing the items at indexes.

When you call this method, Photos first removes the items in the indexes parameter from the collection, and then inserts them at the location specified by the toIndex parameter.
To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before rearranging assets.

12.4.14 moveAssets(fromIndexes as NSIndexSetMBS, toIndex as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the assets at the specified indexes in the asset collection to a new index.
Notes: fromIndexes: The indexes of the assets to be moved in the asset collection.
toIndex: The index at which to place the moved assets, relative to the collection,Äôs ordering after removing the items at indexes.

When you call this method, Photos first removes the items in the indexes parameter from the collection, and then inserts them at the location specified by the toIndex parameter.
To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before rearranging assets.

12.4.15 removeAsset(asset as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the specified asset from the asset collection.
Notes: asset: A PHAssetMBS object to be removed from the asset collection.

This method removes assets from the collection based on their identity (determined by the localIdentifier property of each asset). To remove objects at specified indexes, use the removeAssets method.

Transient asset collections (such as those created with the transientAssetCollectionWithAssets:title: method) do not support adding or removing content.
See also:

- 12.4.16 removeAsset(AtIndex as Integer) 699

12.4.16 removeAsset(AtIndex as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the asset at the specified index from the asset collection.
Notes: index: The index of the asset to be removed from the asset collection.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before removing assets. To remove objects based on their identities (without regard to their indexes in the collection), use the removeAssets: method.

Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.
See also:

- 12.4.15 removeAsset(asset as PHAssetMBS) 699

12.4.17 removeAssets(assets() as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the specified assets from the asset collection.
Notes: assets: An array of PHAssetMBS objects to be removed from the asset collection.

This method removes assets from the collection based on their identity (determined by the localIdentifier property of each asset). To remove objects at specified indexes, use the removeAssets method.

Transient asset collections (such as those created with the transientAssetCollectionWithAssets:title: method) do not support adding or removing content.
See also:

- 12.4.18 removeAssets(AtIndexes as NSIndexSetMBS) 700

12.4.18 removeAssets(AtIndexes as NSIndexSetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the assets at the specified indexes from the asset collection.
Notes: indexes: The indexes of the assets to be removed from the asset collection.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before removing assets. To remove objects based on their identities (without regard to their indexes in the collection), use the removeAssets: method.

Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.
See also:

- 12.4.17 removeAssets(assets() as PHAssetMBS) 700

12.4.19 replaceAsset(AtIndex as Integer, asset as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Replaces the asset at the specified index in the asset collection with the specified asset.
Notes: index: The index of the asset to be replaced in the asset collection.
asset: A PHAssetMBS object to be inserted into (or moved within) the asset collection.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before rearranging assets.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.20 replaceAssets(AtIndexes as NSIndexSetMBS, assets() as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Replaces the assets at the specified indexes in the asset collection with the specified assets.
Notes: indexes: The indexes of the assets to be replaced in the asset collection.
assets: An array of PHAssetMBS objects to be inserted into (or moved within) the asset collection.

To ensure that the index set you specify is valid even if the asset collection has changed since you fetched it, create a change request with a snapshot of the asset collection,Äôs contents using the changeRequestForAssetCollection method before rearranging assets.

Assets from My Photo Stream or iCloud Shared Albums and assets synced to the device through iTunes cannot be added to collections. Transient asset collections (such as those created with the transientAssetCollectionWithAssets method) do not support adding or removing content.

12.4.21 Properties

12.4.22 placeholderForCreatedAssetCollection as PHObjectPlaceholderMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A placeholder object for the asset collection that the change request creates.
Notes: Use this property if you need to reference the asset collection created by a change request within the same change block.
(Read only property)

12.4.23 title as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The displayed name of the asset collection.
Notes: Set this property to change the asset collection,Äôs title.
(Read and Write property)

12.5 class PHAssetCollectionMBS

12.5.1 class PHAssetCollectionMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A representation of a Photos asset grouping, such as a moment, user-created album, or smart album.
Notes: In the Photos framework, collection objects (including asset collections) do not directly reference their member objects, and there are no other objects that directly reference collection objects. To retrieve the members of an asset collection, fetch them with a PHAsset class method such as fetchAssetsInAssetCollection. To find asset collections, use one of the methods listed in Fetching Asset Collections.

Important
Accessing or modifying the Photos library requires explicit authorization from the user. The first time you call one of the methods listed in Fetching Asset Collections, Photos automatically prompts the user for authorization. (Alternatively, you can use the PHPhotoLibraryMBS.requestAuthorization method to prompt the user at a time of your choosing.)

Your app,Äôs Info.plist file must provide a value for the NSPhotoLibraryUsageDescription key that explains to the user why your app is requesting Photos access. Apps linked on or after iOS 10.0 will crash if this key is not present.

Like assets and collection lists, asset collections are immutable. To create, rename, or delete asset collections, or to add, remove, or rearrange members in an asset collection, create a PHAssetCollectionChangeRequestMBS object within a photo library change block. For details on using change requests and change blocks to update the photo library, see PHPhotoLibraryMBS.
Subclass of the PHCollectionMBS class.
Blog Entries

- Show Live Photos in your Xojo application

12.5.2 Methods

12.5.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.5.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.5.5 fetchAssetCollectionsContainingAsset(asset as PHAssetMBS, type as Integer, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves asset collections of the specified type containing the specified asset.
Notes: asset: A Photos asset.
type: An asset collection type, such as an album or a moment. See Type constants.
options: Options that specify a filter predicate and sort order for the fetched asset collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAssetCollectionMBS objects, or an empty fetch result if no objects match the request.

12.5.6 fetchAssetCollectionsWithALAssetGroupURLs(assetGroupURLs() as string, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves asset collections using URLs provided by the Assets Library framework.
Notes: assetGroupURLs: An array of URLs, each an asset group URL that was previously retrieved from an ALAssetsGroupMBS object.
options: Options that specify a filter predicate and sort order for the fetched asset collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAssetCollectionMBS objects, or an empty fetch result if no objects match the request.

The Assets Library framework is deprecated in iOS 8.0 and later, replaced by the Photos framework. Use this method if your app has previously stored URLs from ALAssetsGroupMBS objects and you need to retrieve the corresponding Photos framework objects.

12.5.7 fetchAssetCollectionsWithLocalIdentifiers(identifiers() as string, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves asset collections with the specified unique identifiers.
Notes: identifiers: An array of strings, each the localIdentifier string of an asset collection.
options: Options that specify a filter predicate and sort order for the fetched asset collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAssetCollectionMBS objects, or an empty fetch result if no objects match the request.

12.5.8 fetchAssetCollectionsWithType(type as Integer, subType as integer, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves asset collections of the specified type and subtype.
Notes: type: A type of asset collection, such as an album or a moment. See Type constants.
subtype: A subtype of asset collection. See Subtype constants.
options: Options that specify a filter predicate and sort order for the fetched asset collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAssetCollectionMBS objects, or an empty fetch result if no objects match the request.

By default, the returned PHFetchResultMBS object contains all asset collections with the specified type and subtype. To retrieve a more specific set of asset collections, provide a PHFetchOptionsMBS object containing a filter predicate.

12.5.9 localizedLocationNames as String()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The names of locations grouped by the collection (an array of strings).
Notes: This property applies only to asset collections whose type is PHAssetCollectionTypeMoment. The Photos app automatically creates moments to group assets by time and location. A moment might group assets captured in multiple locations, so this array can contain multiple names. For other asset collection types, this property,Äôs value is nil.

12.5.10 transientAssetCollectionWithAssetFetchResult(fetchResult as PHAssetMBS, title as string) as PHAssetCollectionMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a temporary asset collection containing the assets from the specified fetch result.
Notes: fetchResult: A fetch result containing one or more PHAssetMBS objects.
title: A name for the new temporary asset collection.

Returns a new asset collection.

Transient asset collections are not saved to local storage or iCloud and do not appear in the Photos application or other apps using the Photos framework. A transient collection can be useful if you,Äôve designed a UI for displaying the contents of a collection and want to display an arbitrary set of assets.

12.5.11 transientAssetCollectionWithAssets(assets() as PHAssetMBS, title as string) as PHAssetCollectionMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a temporary asset collection containing the specified assets.
Notes: assets: An array of PHAssetMBS objects.
title: A name for the new temporary asset collection.

Returns a new asset collection.

Transient asset collections are not saved to local storage or iCloud and do not appear in the Photos app or other apps using the Photos framework. A transient collection can be useful if you,Äôve designed a UI for displaying the contents of a collection and want to display an arbitrary set of assets.

12.5.12 Properties

12.5.13 approximateLocation as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A location representing those of all assets in the collection.
Notes: The Photos app automatically creates moments to group assets by time and location. A moment can contain photos from several locations in close proximity. In such cases, this property describes the general area containing the locations of all assets in the moment.
This property applies only to asset collections whose type is TypeMoment. For other asset collection types, this property,Äôs value is nil.

We pass result, the CLLocationMBS object, as variant to avoid plugin dependencies.
(Read only property)

12.5.14 assetCollectionSubtype as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The subtype of the asset collection.
Notes: Use subtypes to make minor distinctions between collections of the same type. For albums, for example, you might distinguish user-created albums from those synced from iPhoto. See Subtype constants for possible values.
(Read only property)

12.5.15 assetCollectionType as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of the asset collection, such as an album or a moment.
Notes: (Read only property)

12.5.16 endDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The latest creation date among all assets in the asset collection.
Notes: This property applies only to asset collections whose type is TypeMoment. For other asset collection types, this property,Äôs value is nil.
(Read only property)

12.5.17 endDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The latest creation date among all assets in the asset collection.
Notes: This property applies only to asset collections whose type is TypeMoment. For other asset collection types, this property,Äôs value is nil.
(Read only property)

12.5.18 estimatedAssetCount as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The estimated number of assets in the asset collection.
Notes: This count may not match the current number of assets in the collection. To get the most recent count, fetch the collection,Äôs assets with the fetchAssetsInAssetCollection method and read the count property of the fetch result.
If asset count information is not available for the collection, this property,Äôs value is NSNotFound.
(Read only property)

12.5.19 startDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The earliest creation date among all assets in the asset collection.
Notes: This property applies only to asset collections whose type is TypeMoment. For other asset collection types, this property,Äôs value is nil.
(Read only property)

12.5.20 startDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The earliest creation date among all assets in the asset collection.
Notes: This property applies only to asset collections whose type is TypeMoment. For other asset collection types, this property,Äôs value is nil.
(Read only property)

12.5.21 Constants

SubTypes

Constant	Value	Description
SubtypeAlbumCloudShared	101	An iCloud Shared Photo Stream.
SubtypeAlbumImported	6	An album imported from a camera or external storage.
SubtypeAlbumMyPhotoStream	100	The user,Äôs personal iCloud Photo Stream.
SubtypeAlbumRegular	2	An album created in the Photos app.
SubtypeAlbumSyncedAlbum	5	An album synced to the device from iPhoto.
SubtypeAlbumSyncedEvent	3	An Event synced to the device from iPhoto.
SubtypeAlbumSyncedFaces	4	A Faces group synced to the device from iPhoto.
SubtypeAny	-1	A bit mask representing all possible subtypes.
SubtypeSmartAlbumAllHidden	205	A smart album that groups all assets hidden from the Moments view i Photos app.
SubtypeSmartAlbumAnimated	214	A smart album that groups all image animation assets.
SubtypeSmartAlbumBursts	207	A smart album that groups all burst photo sequences in the photo librar
SubtypeSmartAlbumDepthEffect	212	A smart album that groups all images captured using the Depth Effect ca mode on compatible devices.
SubtypeSmartAlbumFavorites	203	A smart album that groups all assets that the user has marked as favorit
SubtypeSmartAlbumGeneric	200	A smart album of no more specific subtype.
SubtypeSmartAlbumLivePhotos	213	A smart album that groups all Live Photo assets.
SubtypeSmartAlbumLongExposures	215	A smart album that groups all Live Photo assets where the Long Exp variation is enabled.
SubtypeSmartAlbumPanoramas	201	A smart album that groups all panorama photos in the photo library.
SubtypeSmartAlbumRecentlyAdded	206	A smart album that groups assets that were recently added to the photo lib
SubtypeSmartAlbumScreenshots	211	A smart album that groups all images captured using the device,Äôs scree function.
SubtypeSmartAlbumSelfPortraits	210	A smart album that groups all photos and videos captured using the devic front-facing camera.
SubtypeSmartAlbumSlomoVideos	208	A smart album that groups all Slow-Mo videos in the photo library.
SubtypeSmartAlbumTimelapses	204	A smart album that groups all time-lapse videos in the photo library.
SubtypeSmartAlbumUnableToUpload	216	Unable to upload album.
SubtypeSmartAlbumUserLibrary	209	A smart album that groups all assets that originate in the user,Äôs own li (as opposed to assets from iCloud Shared Albums).
SubtypeSmartAlbumVideos	202	A smart album that groups all video assets in the photo library.

Types

Constant	Value	Description
TypeAlbum	1	An album in the Photos app.
TypeMoment	3	A moment in the Photos app. Deprecated by Apple.
TypeSmartAlbum	2	A smart album whose contents update dynamically.

12.6 class PHAssetCreationRequestMBS

12.6.1 class PHAssetCreationRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A request to create a new Photos asset from underlying data resources, for use in a photo library change block.
Notes: A PHAssetCreationRequestMBS object, used within a photo library change block, constructs a new photo or video asset from data resources, and adds it to the Photos library. This class works in terms of the raw data resources that together form an asset, so you can use it together with the PHAssetResource class to perform a complete copy (or backup and restore) of an asset,Äôs underlying resources. To instead simply create a new asset from an image object, image file, or video file, see the superclass PHAssetChangeRequestMBS.

To create a new asset from data resources, first start a change block using the shared PHPhotoLibraryMBS method performChanges or performChangesAndWait. Then, within the change block:

1. Within the change block, create a new asset creation request with the creationRequestForAsset method.
2. Add image, video, or data resources using the methods listed in Providing Data Resources for the New Asset.
3. (Optional.) Set metadata for the new asset using methods and properties of the superclass PHAssetChangeRequestMBS.

After Photos runs the change block and calls your completion handler, the new asset is created in the Photos library.
If you instantiate or use this class outside a photo library change block, Photos throws an exception. For details on change blocks, see PHPhotoLibrary.
Subclass of the PHAssetChangeRequestMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.6.2 Methods

12.6.3 addResource(type as Integer, data as MemoryBlock, options as PHAssetResourceCreationOptionsMBS = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a data resource to the asset being created, using the specified data.
Notes: type: The role of this data resource in constructing an asset. For details, see type constants.
data: The data for the asset resource.
options: Options affecting how Photos constructs the asset resource and incorporates its data into the Photos library. For details, see PHAssetResourceCreationOptionsMBS class.

Photos imports the asset resource data only when it executes the PHPhotoLibraryMBS change block in which you create a PHAssetCreationRequest object and call this method. If you attempt to create an asset with invalid data or an invalid combination of resources, Photos reports an error in the completion handler of your PHPhotoLibraryMBS call.
See also:

12.6.4 addResource(type as Integer, data as String, options as PHAssetResourceCreationOptionsMBS = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a data resource to the asset being created, using the specified data.
Notes: type: The role of this data resource in constructing an asset. For details, see type constants.
data: The data for the asset resource.
options: Options affecting how Photos constructs the asset resource and incorporates its data into the Photos library. For details, see PHAssetResourceCreationOptionsMBS class.

Photos imports the asset resource data only when it executes the PHPhotoLibraryMBS change block in which you create a PHAssetCreationRequest object and call this method. If you attempt to create an asset with invalid data or an invalid combination of resources, Photos reports an error in the completion handler of your PHPhotoLibraryMBS call.
See also:

12.6.5 addResource(type as Integer, file as FolderItem, options as PHAssetResourceCreationOptionsMBS = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a data resource to the asset being created, using the file at the specified URL.
Notes: type: The role of this data resource in constructing an asset. For details, see type constants.
file: The URL to a local file containing data for the asset resource.
options: Options affecting how Photos constructs the asset resource and incorporates its data into the Photos library. For details, see PHAssetResourceCreationOptionsMBS.

Photos imports the asset resource data only when it executes the PHPhotoLibrary change block in which you create a PHAssetCreationRequestMBS object and call this method. If you attempt to create an asset with invalid data or an invalid combination of resources, Photos reports an error in the completion handler of your PHPhotoLibraryMBS call.
See also:

- 12.6.3 addResource(type as Integer, data as MemoryBlock, options as PHAssetResourceCreationOptionsMBS = nil) 710

- 12.6.4 addResource(type as Integer, data as String, options as PHAssetResourceCreationOptionsMBS = nil) 711

12.6.6 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.6.7 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.6.8 creationRequestForAsset as PHAssetCreationRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new asset to the Photos library using asset resources.
Notes: Call this method within a photo library change block to create a new asset. For details on change blocks, see PHPhotoLibraryMBS. After calling this method, and before returning from the change block use the methods listed in Providing Data Resources for the New Asset to specify one or more data resources for the asset.

To set metadata properties of the newly created asset, use the corresponding properties of the change request (provided by the superclass PHAssetChangeRequestMBS and listed in Modifying Assets). To reference the newly created asset later in the same change block or after the change block completes, use the placeholderForCreatedAsset property to retrieve a placeholder object.

12.6.9 supportsAssetResourceTypes(types() as Integer) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a Boolean value indicating whether Photos supports creating an asset with the specified combination of resource types.
Notes: types: An array of numbers, each the raw value of a PHAssetResourceType identifier.
Return true if Photos supports the specified combination of resource types; otherwise, false.

When you request creation of an asset from resource data, Photos does not validate that the resources can construct a complete asset until the complete PHPhotoLibrary performChanges change block executes. (If an asset cannot be constructed from the provided resources, Photos calls the completionHandler you provide in that method with an error describing the failure.) To perform preflight validation before executing an asset creation request, use this method to verify that the set of resource types from which you want to create an asset are correct.

This method verifies only that the collection of asset resource types is valid (for example, ensuring that you do not attempt to construct a photo asset without image data), so it is still possible for an asset creation request to fail if the data itself is incomplete or invalid. However, calling by using this method you can avoid some kinds of asset creation failure before performing the expensive operation of reading (and potentially downloading or transmitting) asset resource data.

12.6.10 Constants

Resource Types

Constant	Value	Description
TypeAdjustmentBasePairedVideo	11	The resource provides an unaltered version of the video data for a Live Phot asset for use in reconstructing recent edits. Requires MacOS 10.15 or newer.
TypeAdjustmentBasePhoto	8	The resource provides an unaltered version of its photo asset for use in for us in reconstructing recent edits.
TypeAdjustmentBaseVideo	12	The resource provides an unaltered version of the video data for video asset fo use in reconstructing recent edits. Requires MacOS 10.15 or newer.
TypeAdjustmentData	7	The resource provides data for use in reconstructing recent edits to its asset.
TypeAlternatePhoto	4	The resource provides photo data that is not the primary form of its asset.
TypeAudio	3	The resource provides original audio data for its asset.
TypeFullSizePairedVideo	10	The resource provides the current video data component of a Live Photo asset Requires MacOS 10.15 or newer.
TypeFullSizePhoto	5	The resource provides photo data at the highest available size and quality fo its asset.
TypeFullSizeVideo	6	The resource provides video data at the highest available size and quality fo its asset.
TypePairedVideo	9	The resource provides the original video data component of a Live Photo asset
TypePhoto	1	The resource provides original photo data for its asset.
TypeVideo	2	The resource provides original video data for its asset.

12.7 class PHAssetMBS

12.7.1 class PHAssetMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The class representating of an image, video, or Live Photo in the Photos library.
Notes: You fetch assets to begin working with them. Use the class methods listed in Fetching Assets to retrieve one or more PHAsset instances representing the assets you want to display or edit.

Assets contain only metadata. The underlying image or video data for any given asset might not be stored on the local device. However, depending on how you plan to use this data, you may not need to download all of it. If you need to populate a collection view with thumbnail images, the Photos framework can manage downloading, generating, and caching thumbnails for each asset. For details, see PHImageManagerMBS.

Asset objects are immutable. To edit an asset,Äôs metadata (such as marking it as a favorite photo), create a PHAssetChangeRequestMBS object within a photo library change block. For more details on using change requests and change blocks to update the photo library, see PHPhotoLibraryMBS.
Subclass of the PHObjectMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Show Live Photos in your Xojo application

12.7.2 Methods

12.7.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.7.4 cancelContentEditingInputRequest(requestID as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels a request for editing the asset,Äôs content.
Notes: requestID: The numeric identifier of the request to be canceled.

When you request asset editing information with the requestContentEditingInputWithOptions method, the asset returns a numeric identifier for the request. To cancel the request before it completes, provide this

identifier when calling the cancelContentEditingInputRequest method.

12.7.5 canPerformEditOperation(editOperation as Integer) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns whether the asset supports the specified editing operation.
Notes: editOperation: The operation to be tested.

Returns true if the asset supports the the specified editing operation; otherwise, false.

If an asset supports editing, you can create a PHAssetChangeRequestMBS object inside a PHPhotoLibraryMBS change block to submit a change.

12.7.6 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.7.7 fetchAssets(options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves all assets matching the specified options.
Notes: options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS class.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result if no objects match the request.

By default, fetch results do not include photos synced to the device through iTunes or stored in iCloud Shared Albums. To change this behavior, use the includeAssetSourceTypes property in the options parameter.

12.7.8 fetchAssetsInAssetCollection(assetCollection as PHAssetCollectionMBS, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves assets from the specified asset collection.
Notes: assetCollection: The asset collection from which to fetch assets.
options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS class.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result if no objects match the request.

By default, the returned PHFetchResultMBS object contains all assets in the specified collection. To retrieve a more specific set of assets, provide a PHFetchOptionsMBS object containing a filter predicate.

12.7.9 fetchAssetsWithBurstIdentifier(burstIdentifier as String, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves assets with the specified burst photo sequence identifier.
Notes: burstIdentifier: A burst identifier string, as provided by the burstIdentifier property of an asset.
options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result if no objects match the request.

A burst photo sequence, as seen in the Photos app, corresponds to a group of Photos assets that share the same burstIdentifier string.

By default, the returned PHFetchResultMBS object contains only the representative asset and any user-picked photos from the burst sequence. To retrieve all photos in the burst sequence, provide a PHFetchOptionsMBS object containing a filter predicate.

12.7.10 fetchAssetsWithLocalIdentifiers(identifiers() as String, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves assets with the specified local-device-specific unique identifiers.
Notes: identifiers: An array of strings, each the localIdentifier string of an asset.
options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS class.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result if no objects match the request.

12.7.11 fetchAssetsWithMediaType(mediaType as Integer, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves assets with the specified media type.
Notes: mediaType: A media type, such as image or video. See type constants.
options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result if no objects match the request.

By default, the returned PHFetchResultMBS object contains all assets with the specified type. To retrieve a more specific set of assets, provide a PHFetchOptionsMBS object containing a filter predicate.

By default, fetch results do not include photos synced to the device through iTunes or stored in iCloud Shared Albums. To change this behavior, use the includeAssetSourceTypes property in the options parameter.

12.7.12 fetchKeyAssetsInAssetCollection(assetCollection as PHAssetCollectionMBS, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves assets marked as key assets in the specified asset collection.
Notes: assetCollection: The asset collection from which to fetch assets.
options: Options that specify a filter predicate and sort order for the fetched assets, or nil to use default options. For details, see PHFetchOptionsMBS class.

Returns a fetch result that contains the requested PHAsset objects, or an empty fetch result or nil if no objects match the request.

Most asset collections contain a key asset, which the Photos app displays as a proxy for the collection. Different types of asset collections have different ways of specifying one or more key assets. For example, in the Camera Roll collection, the most recently captured photo or video is the key asset.

This method returns nil if the assetCollection parameter references a transient asset collection (such as one created with the transientAssetCollectionWithAssets method).

12.7.13 PHContentEditingInputCancelledKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys indicating the status of an asset content editing request.
Notes: A Boolean value indicating whether the image request was canceled. (boolean)

If you call the cancelContentEditingInputRequest method to cancel a request, Photos calls your result handler delegate with the value true for this key.

12.7.14 PHContentEditingInputErrorKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys indicating the status of an asset content editing request.
Notes: An error that occurred while attempting to load the asset data, NSErrorMBS.

Photos provides an error object for this key if it cannot provide asset data for your handler block,Äôs contentEditingInput parameter. Examine the error object for information about the cause of the error.

12.7.15 PHContentEditingInputResultIsInCloudKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys indicating the status of an asset content editing request.
Notes: A Boolean value indicating whether the asset data is stored on the local device or must be downloaded from iCloud.

If true, no asset data was provided because the asset data must be downloaded from iCloud. To do this, submit another request, specifying true for the networkAccessAllowed option.

12.7.16 requestContentEditingInputWithOptions(options as PHContentEditingInputRequestOptionsMBS, CompletionDelegate as RequestContentEditingInputWithOptionsCompletedMBS, tag as variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests asset information for beginning a content editing session.
Notes: options: Options affecting how Photos handles an edit session request.

Returns a numeric identifier for the request. Pass this identifier to the cancelContentEditingInputRequest method if you need to cancel the request before it completes.

When you call this method, Photos downloads the asset,Äôs image or video data (if necessary) and prepares it for editing, then calls your completionHandler ddelegate to provide a PHContentEditingInputMBS object you use for editing.

To complete the edit, create a PHContentEditingOutputMBS object from the editing input to provide the edited asset data. Then, commit the edit by posting a change block to the shared PHPhotoLibrary object. In the block, create a PHAssetChangeRequestMBS object and set its contentEditingOutput property to the editing output you created.

12.7.17 Properties

12.7.18 burstIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The unique identifier shared by photo assets from the same burst sequence.
Notes: When the user takes a sequence of photos in burst mode with the Camera app (on supported devices), the Photos app user interface groups the resulting assets together. The Photos framework identifies a burst sequence as a group of assets sharing the same burst identifier string.
(Read only property)

12.7.19 burstSelectionTypes as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The selection type of the asset in a burst photo sequence.
Notes: When the user takes a sequence of photos in burst mode with the Camera app (on supported devices), the Photos app user interface groups the resulting assets together and allows the user to select favorite members of the sequence. Photos also automatically marks members of the sequence as potential user favorites. See PHAssetBurstSelectionType for possible values.

Because an asset may have more than one selection type, you use bit masks to identify an asset.
(Read only property)

12.7.20 creationDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The date and time at which the asset was originally created.
Notes: (Read only property)

12.7.21 creationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The date and time at which the asset was originally created.
Notes: (Read only property)

12.7.22 duration as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The duration, in seconds, of the video asset.
Notes: For photo assets, the duration is always zero.
(Read only property)

12.7.23 Favorite as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the user has marked the asset as a favorite.
Notes: (Read only property)

12.7.24 Hidden as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the user has hidden the asset.
Notes: Hidden assets do not appear in the Moments interface in the Photos app and are not returned when fetching the contents of a moment collection.
(Read only property)

12.7.25 location as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The location information saved with the asset.
Notes: Typically, an asset,Äôs location metadata identifies the place where the asset was captured.

We pass result, the CLLocationMBS object, as variant to avoid plugin dependencies.
(Read only property)

12.7.26 mediaSubtypes as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The subtypes of the asset, identifying special kinds of assets such as panoramic photo or high-framerate video.
Notes: See subtype constants for possible values.
Because an asset may have more than one subtype, you use these values as bit masks to identify an asset.
(Read only property)

12.7.27 mediaType as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of the asset, such as video or audio.
Notes: See Type constants.
(Read only property)

12.7.28 modificationDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The date and time at which the asset was last modified.
Notes: Photos updates the modification date when an asset,Äôs image or video content or metadata changes.
(Read only property)

12.7.29 modificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The date and time at which the asset was last modified.
Notes: Photos updates the modification date when an asset,Äôs image or video content or metadata changes.
(Read only property)

12.7.30 pixelHeight as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The height, in pixels, of the asset,Äôs image or video data.
Notes: If the asset,Äôs content has been edited, this property describes the size of the current version of the asset.
(Read only property)

12.7.31 pixelWidth as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The width, in pixels, of the asset,Äôs image or video data.
Notes: If the asset,Äôs content has been edited, this property describes the size of the current version of the asset.
(Read only property)

12.7.32 playbackStyle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The playback style for the asset.
Notes: (Read only property)

12.7.33 RepresentsBurst as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the asset is the representative photo from a burst photo sequence.
Notes: When the user takes a sequence of photos in burst mode with the Camera app (on supported devices), the Photos app user interface groups the resulting assets together. One asset represents the entire sequence in displayed collections.
(Read only property)

12.7.34 sourceType as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The means by which the asset entered the user,Äôs Photos library.
Notes: Assets from some sources can be edited, deleted, or added to collections; assets from other sources do not support these operations. For details, see SourceType constants.
(Read only property)

12.7.35 Constants

Constants

Constant	Value	Description
MediaSubtypeNone	0	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset has no subtype. This is the default subtype for most photo and video assets.
MediaSubtypePhotoDepthEffect	16	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a photo captured with the Camera app's Portrait mode effect.
MediaSubtypePhotoHDR	2	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a High Dynamic Range photo.
MediaSubtypePhotoLive	8	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a Live Photo that includes movement and sounds from the mo just before and after its capture. To display a Live Photo asset with its associated video content, retr PHLivePhotoMBS object using the PHImageManagerMBS class and as to a PHLivePhotoViewMBS object.
MediaSubtypePhotoPanorama	1	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a large-format panorama photo.
MediaSubtypePhotoScreenshot	4	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is an image captured with the device,Äôs screenshot feature.
MediaSubtypeVideoHighFrameRate	&h20000	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a high-frame-rate video. High-frame-rate videos are created by the Slow-Mo feature in the Came on an iOS device.
MediaSubtypeVideoStreamed	&h10000	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a video whose contents are always streamed over a networ nection. This subtype identifies video assets that are never stored on the local d such as shared videos in a subscribed iCloud Photo Stream.
MediaSubtypeVideoTimelapse	&h40000	One of the constants identifying specific variations of asset media, su panorama or screenshot photos and time lapse or high frame rate video. The asset is a time-lapse video.

Burst Selection Type

Constant	Value	Description
BurstSelectionTypeAutoPick	1	Photos has automatically identified the asset as a potential user favorite.
BurstSelectionTypeNone	0	The asset is not marked as a favorite member of its burst sequence or is not a member of a burst sequence.
BurstSelectionTypeUserPick	2	The user has marked the asset as a favorite member of its burst sequence.

Edit Operations

Constant	Value	Description
EditOperationContent	2	The asset,Äôs photo or video content can be edited.
EditOperationDelete	1	The asset can be deleted from the photo library.
EditOperationProperties	3	The asset,Äôs metadata properties can be edited.

Media Types

Constant	Value	Description
MediaTypeAudio	3	The asset is an audio file.
MediaTypeImage	1	The asset is a photo or other static image.
MediaTypeUnknown	0	The asset,Äôs type is unknown.
MediaTypeVideo	2	The asset is a video file.

Asset Playback Styles

Constant	Value	Description
PlaybackStyleImage	1	The asset should be displayed as a still image.
PlaybackStyleImageAnimated	2	The asset should be displayed as an animated image.
PlaybackStyleLivePhoto	3	The asset should be displayed as a Live Photo.
PlaybackStyleUnsupported	0	The asset has an unsupported or undefined media playback type.
PlaybackStyleVideo	4	The asset should be displayed as a video.
PlaybackStyleVideoLooping	5	The asset should be displayed as a looping video.

Source Types

Constant	Value	Description
SourceTypeCloudShared	2	The asset originates from an iCloud Shared Album. Assets from shared albums cannot be edited and do not appear in Moments collections.
SourceTypeiTunesSynced	4	The asset originates from a Mac or PC and is present on the device through iTunes sync. iTunes-synced assets cannot be edited or deleted.
SourceTypeNone	0	Source information is not available for the asset.
SourceTypeUserLibrary	1	The asset is part of the user,Äôs main Photos library. The main library contains both assets that originate on the device (such as photos and videos captured with the Camera app or screenshots) and assets synchronized through iCloud Photo Library or My Photo Stream. These assets appear in Moments collections and can be edited or deleted.

12.7.36 Delegates

12.7.37 RequestContentEditingInputWithOptionsCompletedMBS(Asset as PHAssetMBS, Options as PHContentEditingInputRequestOptionsMBS, tag as variant, contentEditingInput as PHContentEditingInputMBS, info as Dictionary)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls when the requested asset editing information is ready.
Notes: contentEditingInput: An object that describes the asset for editing and provides methods for loading the image or video content to be edited.
info: A dictionary providing information about the status of the request. See Editing Request Info Keys for possible keys and values.

12.8 class PHAssetResourceCreationOptionsMBS

12.8.1 class PHAssetResourceCreationOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the creation of a new Photos asset from underlying resources.
Notes: You use this class when creating an asset for addition to the Photos library with a PHAssetCreationRequestMBS object.

12.8.2 Methods

12.8.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.8.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.8.5 copy as PHAssetResourceCreationOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Copies the options object.

12.8.6 Properties

12.8.7 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.8.8 originalFilename as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The filename for the asset resource being created.
Notes: You can use this property to track the original name of the file from which you import an asset resource even if you use the addResourceWithType method to create a resource from data instead of from a file. After creating the asset, this information is available in the originalFilename property of the corresponding PHAssetResourceMBS object.

If you do not specify a value for this property and are using the addResourceWithType method to create a resource, Photos infers the filename from that method,Äôs fileURL parameter. Otherwise, Photos automatically generates a filename.
(Read and Write property)

12.8.9 shouldMoveFile as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether Photos moves or duplicates files when creating an asset resource.
Notes: This property applies only when creating an asset resource with the addResourceWithType method. If this value is true, Photos moves the specified file into the Photos library to create the asset resource, removing the original file after the asset has been successfully created. When using this option, Photos does not make an intermediary copy of the resource data, so no additional storage space is required.

If this value is false (the default), Photos copies the contents of the original file into the Photos library.

Attempting to move a file that is currently open or has hard links fails.
(Read and Write property)

12.8.10 uniformTypeIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The uniform type identifier for the resource.
Notes: If you do not specify a value for this property, Photos infers the data type from the type value you specify when adding the resource to a creation request.
For details in uniform type identifiers, see Uniform Type Identifiers Overview.
(Read and Write property)

12.9 class PHAssetResourceManagerMBS

12.9.1 class PHAssetResourceManagerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A resource manager for the data storage underlying a Photos asset.
Notes: An asset can have multiple underlying data resources—for example, both original and edited versions—each of which is represented by a PHAssetResourceMBS object. Unlike the PHImageManagerMBS class, which provides and caches the primary representations of assets as thumbnails, image objects, or video objects, the asset resource manager provides direct access to these underlying data resources.

Available in MacOS 10.15 or newer.

12.9.2 Methods

12.9.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.9.4 cancelDataRequest(requestID as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels an asynchronous request.
Notes: requestID: The numeric identifier of the request to be canceled.

When you perform an asynchronous request for asset resource data using the requestDataForAssetResource method, the image manager returns a numeric identifier for the request. To cancel the request before it completes, provide this identifier when calling the cancelDataRequest method.

12.9.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.9.6 defaultManager as PHAssetResourceManagerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the shared asset resource manager object.
Notes: This method always returns the same asset resource manager object, which is shared for all uses in your app.

12.9.7 requestDataForAssetResource(resource as PHAssetResourceMBS, options as PHAssetResourceRequestOptionsMBS, DataReceivedHandler as RequestDataForAssetResourceDataReceivedMBS, CompleteHandler as RequestDataForAssetResourceCompletedMBS, tag as variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests the underlying data for the specified asset resource, to be delivered asynchronously.
Notes: resource: The asset resource for which to request data.
options: Options specifying how Photos should handle the request and notify your app of progress. For details, see PHAssetResourceRequestOptionsMBS.

Returns a numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelDataRequest: method.

When you call this method, Photos begins asynchronously reading the underlying data for the asset resource. Depending on the options you specify and the current state of the asset, Photos may download asset data from the network.

While reading (or downloading) asset resource data, Photos calls your handler block at least once, progressively providing chunks of data. After reading all of the data, Photos calls your completionHandler block to indicate that the data is complete. (At this point, the complete data for the asset is the concatenation of the data parameters from all calls to your handler block.) If Photos cannot finish reading or downloading asset resource data, it calls your completionHandler block with a description of the error. Photos can also call the completionHandler block with a non-nil error when the data is complete if the user cancels downloading.

12.9.8 writeDataForAssetResource(resource as PHAssetResourceMBS, file as FolderItem, options as PHAssetResourceRequestOptionsMBS, CompleteHandler as WriteDataForAssetResourceCompletedMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests the underlying data for the specified asset resource, to be asynchronously written to a local file.
Notes: resource: The asset resource for which to request data.
fileURL: A URL identifying the local filename at which to write the asset resource,Äôs data.
options: Options specifying how Photos should handle the request and notify your app of progress. For details, see PHAssetResourceRequestOptionsMBS.

When you call this method, Photos begins asynchronously reading the underlying data for the asset resource. Depending on the options you specify and the current state of the asset, Photos may download asset data from the network.

While reading (or downloading) asset resource data, Photos progressively writes the data into the specified file. After writing all of the data, or if an error prevents reading all of the data, Photos calls your delegate.

12.9.9 Properties

12.9.10 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.9.11 Constants

Constants

Constant	Value	Description
PHInvalidAssetResourceDataRequestID	0	The invalid request ID.

12.9.12 Delegates

12.9.13 RequestDataForAssetResourceCompletedMBS(resource as PHAssetResourceMBS, options as PHAssetResourceRequestOptionsMBS, Error as NSErrorMBS, Tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that photos calls after the request has been fulfilled or has failed.
Notes: error: If the request has failed, an NSErrorMBS object describing the failure; otherwise nil.

12.9.14 RequestDataForAssetResourceDataReceivedMBS(resource as PHAssetResourceMBS, options as PHAssetResourceRequestOptionsMBS, Data as MemoryBlock, Tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls to provide the requested data.
Notes: data: The requested asset resource data.

12.9.15 WriteDataForAssetResourceCompletedMBS(resource as PHAssetResourceMBS, options as PHAssetResourceRequestOptionsMBS, File as FolderItem, Error as NSErrorMBS, Tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that photos calls after the request has been fulfilled or has failed.
Notes: error: If the request has failed, an NSErrorMBS object describing the failure; otherwise nil.

12.10 class PHAssetResourceMBS

12.10.1 class PHAssetResourceMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An underlying data resource associated with a photo, video, or Live Photo asset in the Photos library.
Notes: Each PHAssetMBS object references one or more resources. Use these objects to work with those resources directly, like when backing up or restoring assets.

- A photo asset can contain both JPEG and RAW files representing the same photo.
- A Live Photo asset contains both still photo and video resources.
- An edited asset contains resources representing asset content before and after the edit, as well as a resource corresponding to the PHAdjustmentDataMBS object that describes the edit.

To work with the data contained in an asset resource, fetch it using the PHAssetResourceManagerMBS class. This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.10.2 Methods

12.10.3 assetResourcesForAsset(asset as PHAssetMBS) as PHAssetResourceMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the list of data resources associated with an asset.
Notes: asset: A photo or video asset in the Photos library.

Returns the asset,Äôs resources.

Asset resource objects describe the data files that an asset represents. An asset can contain multiple resources—for example, an edited photo asset contains resources for both the original and edited images, as well as for the PHAdjustmentDataMBS object describing the edit. To work with one of these files, fetch the underlying data using the PHAssetResourceManagerMBS class.

12.10.4 assetResourcesForLivePhoto(livePhoto as PHLivePhotoMBS) as PHAssetResourceMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the list of data resources associated with a Live Photo object.
Notes: livePhoto: A Live Photo object.

Returns the underlying resources that constitute the Live Photo.

A Live Photo is a picture, taken with a compatible device, that includes movement and sound from the moments just before and after its capture. A PHLivePhoto object represents the displayable combination of image, motion, and sound data. You can obtain such objects from the Photos library using the PHImageManagerMBS class or construct them from asset resources exported from a Photos library using the PHLivePhotoMBS class.

Use this method to export the underlying resources that constitute a Live Photo. For example, a social networking app can retrieve those data files and upload them to a server. Then, on another user,Äôs device, the app downloads those data files and uses the PHLivePhoto class to re-create a Live Photo object for display using the PHLivePhotoViewMBS class.

12.10.5 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.10.6 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.10.7 Properties

12.10.8 assetLocalIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The unique identifier for the asset object this resource is associated with.
Notes: This property,Äôs value corresponds to the localIdentifier property of the PHAssetMBS object that owns this asset resource. If you,Äôve obtained an asset resource without a reference to its owning asset, use the fetchAssetsWithLocalIdentifiers method with this identifier to retrieve the correct PHAssetMBS object. (Read only property)

12.10.9 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.10.10 originalFilename as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The original filename of the asset resource from when it was created or imported.
Notes: (Read only property)

12.10.11 Type as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The relationship of this asset resource to its owning asset.
Notes: An asset can contain multiple resources, and different resources contribute to the asset in different ways. For details and examples, see resource types.
(Read only property)

12.10.12 uniformTypeIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The uniform type identifier for the asset resource,Äôs image or video data.
Notes: For more information, see Uniform Type Identifiers Overview in Apple's documentation.
(Read only property)

12.11 class PHAssetResourceRequestOptionsMBS

12.11.1 class PHAssetResourceRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the delivery of underlying asset data that you request from the asset resource manager.
Notes: You use this class when requesting the underlying data for photo, video, and Live Photo asset resources from a PHAssetResourceManagerMBS object.

Requires MacOS 10.15 or newer.

12.11.2 Methods

12.11.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.11.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.11.5 copy as PHAssetResourceRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the request options.

12.11.6 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.11.7 SetProgressHandler(ProgressHandler as ProgressHandlerMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the delegate that Photos calls periodically while downloading the asset resource data.
Notes: If you request an asset resource whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress.

12.11.8 Properties

12.11.9 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.11.10 NetworkAccessAllowed as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether Photos can download the requested asset resource data from iCloud.
Notes: If true, and the requested resource data is not stored on the local device, Photos downloads that data from iCloud. To be notified of the download,Äôs progress, use the progressHandler property to provide a block that Photos calls periodically while downloading the resource data. If false (the default), and the resource data is not on the local device, Photos calls the completionHandler block you provided in your request, with an NSErrorMBS object indicating that the resource requires network access.
(Read and Write property)

12.11.11 ProgressHandler as ProgressHandlerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls periodically while downloading the asset resource data.
Notes: If you request an asset resource whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress.
(Read and Write property)

12.11.12 Delegates

12.11.13 ProgressHandlerMBS(progress as double, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls while downloading asset resource data from iCloud. Used by the progressHandler property.
Notes: If you request an asset resource whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress.

progress: A floating-point value indicating the progress of the download. A value of 0.0 indicates that the download has just started, and a value of 1.0 indicates the download is complete.

12.12 class PHCachingImageManagerMBS

12.12.1 class PHCachingImageManagerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that facilitates retrieving or generating preview thumbnails, optimized for batch preloading large numbers of assets.
Notes: For quick performance when you are working with many assets, a caching image manager can prepare asset images in the background in order to eliminate delays when you later request individual images. For example, use a caching image manager when you want to populate a collection view or similar UI with thumbnails of photo or video assets.

Much of the key functionality of the PHCachingImageManager class is defined by its superclass, PHImageManagerMBS. For details, see PHImageManagerMBS.

To use a caching image manager:
1. Create a PHCachingImageManager instance. (This step replaces using the shared PHImageManager instance.)
2. Use PHAsset class methods to fetch the assets you,Äôre interested in.
3. To prepare images for those assets, call the startCachingImagesForAssets:targetSize:contentMode:options: method with the target size, content mode, and options you plan to use when later requesting images for each individual asset.
4. When you need an image for an individual asset, call the requestImageForAsset:targetSize:contentMode:options:resultHandler: method, and pass the same parameters you used when preparing that asset.

If the image you request is among those already prepared, the PHCachingImageManager object immediately returns that image. Otherwise, Photos prepares the image on demand and caches it for later use.
Subclass of the PHImageManagerMBS class.

12.12.2 Methods

12.12.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.12.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.12.5 startCachingImagesForAssets(assets() as PHAssetMBS, targetSize as CGSizeMBS, contentMode as Integer, options as PHImageRequestOptionsMBS = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function:

Prepares image representations of the specified assets for later use.

Notes:

assets: An array of PHAssetMBS objects for which to prepare image representations.
targetSize: The size of the images to be prepared.
contentMode: An option for how to fit the images to the aspect ratio of the requested size. For details, see PHImageContentMode.
options: Options specifying how Photos should handle the request, format the requested images, and notify your app of progress or errors. For details, see PHImageRequestOptions.

When you call this method, Photos begins to fetch image data and generates thumbnail images on a background thread. At any time afterward, you can use the requestImageForAsset:targetSize method to request individual images from the cache. If Photos has finished preparing a requested image, that method provides the image immediately.

Photos caches images with the exact target size, content mode, and options you specify in this method. If you later request an image with, for example, a different target size than you passed when calling this method, Photos cannot make use of the cache and so it must fetch or generate a new image.

12.12.6 stopCachingImagesForAllAssets

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels all image preparation that is currently in progress.

12.12.7 stopCachingImagesForAssets(assets() as PHAssetMBS, targetSize as CGSizeMBS, contentMode as Integer, options as PHImageRequestOptionsMBS = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels image preparation for the specified assets and options.
Notes: assets: The array of specific PHAssetMBS objects for which image preparation is in progress but is no longer needed.
targetSize: The target size with which you requested image preparation.
contentMode: The content mode with which you requested image preparation.
options: The options with which you requested image preparation.

This method cancels image preparation for the specified assets with the specified options. Use it when image preparation that might be in progress is no longer needed. For example, if you prepare images for a collection view filled with photo thumbnails and then the user chooses a different thumbnail size for your collection view, call this method to cancel generating thumbnail images at the old size.

12.12.8 Properties

12.12.9 allowsCachingHighQualityImages as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the image manager prepares high-quality images.
Notes: If true (the default), the image manager prepares images at high quality. This option produces better images, at a high performance cost.
For faster performance when preparing large numbers of images—such as while the user is scrolling quickly through a collection of thumbnails—set this property to false.
(Read and Write property)

12.13 class PHChangeMBS

12.13.1 class PHChangeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A description of a change that occurred in the photo library.
Notes: Photos provides PHChangeMBS objects to notify your app of changes to the assets and collections managed by the Photos app. To receive change information, implement the DidChange event in your PHPhotoLibraryMBS subclass.

After Photos provides you with a change object, you use its methods to get a change details object. Call the changeDetailsForObject or changeDetailsForFetchResult method, passing an asset or collection object you,Äôve previously fetched or a fetch result containing several such objects. The resulting PHObjectChangeDetailsMBS or PHFetchResultChangeDetailsMBS object describes any changes that have happened to the object or fetch result since you last fetched it.

12.13.2 Methods

12.13.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.13.4 changeDetailsForFetchResult(FetchResult as PHFetchResultMBS) as PHFetchResultChangeDetailsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns detailed change information for a fetch result.
Notes: object: A fetch result.

Returns a change details object, or nil if there have been no changes affecting the fetch result,Äôs contents.

When Photos calls your change observer,Äôs photoLibraryDidChange method, call the changeDetailsForObject method to get detailed change information about the results of a fetch you,Äôve previously performed. If there have been any changes in the Photos library affecting the fetch, the resulting PHFetchResultChangeDetailsMBS object tells you if any contents in the fetch result have been added, removed, or changed since you fetched it. If there have been no changes since you performed the fetch, this method returns nil.

Typically, if your app displays the members of a collection (such as an album or moment), you use a method such as fetchAssetsInAssetCollection to retrieve those members and then keep the resulting PHFetchResultMBS object. You can then pass that fetch result to this method to learn about changes to the collection, such as whether new members have been added to it (and which indexes to insert them at in your UI).

To find out about changes to an object,Äôs properties, such as a collection,Äôs title or an asset,Äôs metadata, use the changeDetailsForObject: method.

12.13.5 changeDetailsForObject(PHObject as PHObjectMBS) as PHObjectChangeDetailsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns detailed change information for the specified asset or collection.
Notes: object: A PHAssetMBS, PHAssetCollectionMBS, or PHCollectionListMBS object.

Returns a change details object, or nil if there have been no changes to the specified asset or collection.

When Photos calls your change observer,Äôs photoLibraryDidChange: method, call the changeDetailsForObject method to get detailed change information about an asset or collection you,Äôve previously fetched. If the asset or collection has changed since you last fetched it, the resulting PHObjectChangeDetailsMBS object describes the changes. If there are no changes between the fetched object and the current state of the asset or collection it represents in the Photos library, this method returns nil.
For an asset collection or collection list, this method and the PHObjectChangeDetailsMBS object it returns describe changes only to the collection,Äôs properties. If you are instead interested in changes to the collection,Äôs membership, use the changeDetailsForFetchResult method.

12.13.6 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.13.7 Properties

12.13.8 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.14 class PHChangeRequestMBS

12.14.1 class PHChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A request to create, delete, change metadata for, or edit the content of a Photos asset, for use in a photo library change delegate.
Notes: You use the PHAssetChangeRequestMBS class to request changes for PHAssetMBS objects. To make changes to assets in the Photos library, create a change request by using the appropriate class method for the change you want to perform.

- Call one of the methods listed in Adding New Assets to create a new asset from an image or video file.
- Call the deleteAssets: method to delete existing assets.
- Call the changeRequestForAsset: method to modify an asset,Äôs content or metadata.

A change request for creating or modifying an asset works like a mutable version of the asset object. Use the change request,Äôs properties to request changes to the corresponding properties of the asset itself.

After Photos runs the change block and calls your completion handler, the asset,Äôs state reflects the changes that you requested in the block.

If you create or use a change request object outside a photo library change block, Photos raises an Objective-C exception. For details on change blocks, see PHPhotoLibraryMBS.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.14.2 Methods

12.14.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.14.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.14.5 Properties

12.14.6 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.15 class PHCloudIdentifierMBS

12.15.1 class PHCloudIdentifierMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A cloud identifier for a Photos project extension.
Notes: The user can choose to store Photos project extensions in iCloud. The cloud identifier uniquely references project extensions in the cloud by string serialization. When a project's global identifier can't be determined from its local identifier, the notFoundIdentifier identifier is provided in that array slot.

12.15.2 Methods

12.15.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.15.4 Constructor(stringValue as String)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Deserializes a cloud identifier from its string value.

12.15.5 notFoundIdentifier as PHCloudIdentifierMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The global identifier used in an array slot for items that couldn't be found.

12.15.6 Properties

12.15.7 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.15.8 stringValue as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A string version of the cloud identifier to use in serialization.
Notes: (Read only property)

12.16 class PHCollectionListChangeRequestMBS

12.16.1 class PHCollectionListChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A request to create, delete, or modify a Photos collection list, for use in a photo library change block.
Notes: You use the PHCollectionListChangeRequest class to request changes for PHCollectionList objects. To make changes to collection lists (such as folders containing user-created albums) in the Photos library, create a change request using the appropriate class method for the change you want to perform.

- Call the creationRequestForCollectionListWithTitle method to create a new asset collection.
- Call the deleteCollectionLists: method to delete existing asset collections.
- Call the changeRequestForCollectionList: or changeRequestForCollectionList method to modify a collection,Äôs metadata or its list of child collections.

Before creating a change request, use the canPerformEditOperation: method to verify that the collection allows the edit operation you,Äôre requesting. If you attempt to perform an unsupported edit operation, Photos throws an exception.
A change request for creating or modifying a collection list works like a mutable version of the collection list object. Use the change request,Äôs properties and instance methods to request changes to the collection list itself.

After Photos runs the change delegate and calls your completion handler, the collection list,Äôs state reflects the changes you requested in the delegate.

If you create or use a change request object outside a photo library change delegate, Photos raises an Objective-C exception. For details on change delegates, see PHPhotoLibraryMBS.
Subclass of the PHChangeRequestMBS class.

12.16.2 Methods

12.16.3 addChildCollections(collections() as PHCollectionMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Add the specified collections as children of the collection list.
Notes: collections: An array of PHCollectionMBS objects (asset collections or other collection lists) to be added to the collection list.

If you created the change request with a snapshot of the collection list,Äôs contents using the changeRequestForCollectionList method, Photos inserts the new children after the existing child collections in the collection list. Otherwise, the arrangement of the new children relative to others in the collection is undefined.

12.16.4 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.16.5 changeRequestForCollectionList(collectionList as PHCollectionListMBS) as PHCollectionListChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for modifying the specified collection list.
Notes: collectionList: The collection list to be modified.

Returns a collection list change request.

After you create a change request within a photo library change block, propose changes to the collection,Äôs title or list of child collections with the properties and instance methods of the change request. After Photos runs your change block, the collection list reflects your changes. For details on change blocks, see PHPhotoLibraryMBS.
Use this method when modifying a collection list,Äôs metadata or when adding or removing child collections without regard to their arrangement. To work with indexes in the list of child collections, use the changeRequestForCollectionList method instead.
See also:

- 12.16.6 changeRequestForCollectionList(collectionList as PHCollectionListMBS, childCollections as PHFetchResultMBS) as PHCollectionListChangeRequestMBS 751

12.16.6 changeRequestForCollectionList(collectionList as PHCollectionListMBS, childCollections as PHFetchResultMBS) as PHCollectionListChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for modifying the specified collection list, with a fetch result for tracking changes.
Notes: collectionList: The collection list to be modified.
childCollections: A fetch result listing the child collections in the collection.

Returns a collection list change request.

After you create a change request within a photo library change block, propose changes to the collection,Äôs title or list of child collections with the properties and instance methods of the change request. After Photos runs your change block, the collection list reflects your changes. For details on change blocks, see PHPhotoLibraryMBS.

Use this method when you need to insert, remove, or rearrange collections at specified indexes in the collection list. By passing in a fetch result reflecting what your app sees as the current state of the collection,Äôs membership, the Photos framework can ensure that the indexes you specify are valid even if the collection has changed since you last fetched it. If you don,Äôt need to work with indexes in the list of child collections, you can use the changeRequestForCollectionList method instead.
See also:

- 12.16.5 changeRequestForCollectionList(collectionList as PHCollectionListMBS) as PHCollectionListChangeRequestMBS 751

12.16.7 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.16.8 creationRequestForCollectionListWithTitle(title as string) as PHCollectionListChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a request for adding a new collection list to the Photos library.
Notes: title: A name for the new collection list.

Returns a collection list creation request.

Call this method within a photo library change block to create a new collection list. For details on change blocks, see PHPhotoLibraryMBS.
To add collections to the newly created collection list or to change its title, use the methods listed in Modifying Collection Lists. To reference the newly created collection collection list later in the same change delegate or after the change delegate completes, use the placeholderForCreatedCollectionListMBS property to retrieve a placeholder object.

12.16.9 deleteCollectionLists(collectionLists() as PHCollectionListMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests to delete the specified asset collections.
Notes: collectionLists: An array of PHCollectionListMBS objects to be deleted.

Call this method within a photo library change block to delete collection lists. For details on change blocks, see PHPhotoLibraryMBS.

Deleting a collection list also deletes any child collections it contains. To preserve those collections, remove them from the collection list (with the removeChildCollections or removeChildCollectionsAtIndexes method) before deleting it. Deleting a collection list does not delete assets contained in its child collections.

12.16.10 insertChildCollections(collections() as PHCollectionMBS, indexes as NSIndexSetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Inserts the specified collections into the collection list at the specified indexes.
Notes: collections: An array of PHCollectionMBS objects (asset collections or other collection lists) to be inserted into the collection list.
indexes: The indexes at which the collections should be inserted. The count of locations in this index set must equal the count of collections.

To ensure that the index set you specify is valid even if the collection list has changed since you fetched it, create a change request with a snapshot of the collection list,Äôs contents using the changeRequestForCollectionList method before inserting child collections.
For a detailed discussion of how the index set you specify maps to insertions in the collection list, see the similar NSMutableArray method insertObjects.

12.16.11 moveChildCollections(AtIndexes as NSIndexSetMBS, toIndex as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the child collections at the specified indexes in the collection list to a new index.
Notes: indexes: The indexes of the child collections to be moved in the collection list.
toIndex: The index at which to place the moved child collections, relative to the collection list,Äôs ordering after removing the items at indexes.

When you call this method, Photos first removes the items in the indexes parameter from the collection, and

then inserts them at the location specified by the toIndex parameter.
To ensure that the index set you specify is valid even if the collection list has changed since you fetched it, create a change request with a snapshot of the collection list,Äôs contents using the changeRequestForCollectionList method before rearranging child collections.

12.16.12 removeChildCollections(AtIndexes as NSIndexSetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the child collections at the specified indexes from the collection list.
Notes: indexes: The indexes of the child collections to be removed from the collection list.

To ensure that the index set you specify is valid even if the collection list has changed since you fetched it, create a change request with a snapshot of the collection list,Äôs contents using the changeRequestForCollectionList method before removing child collections. To remove objects based on their identities (without regard to their indexes in the collection), use the removeChildCollections method.
See also:

- 12.16.13 removeChildCollections(collections() as PHCollectionMBS) 754

12.16.13 removeChildCollections(collections() as PHCollectionMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the specified child collections from the collection list.
Notes: collections: An array of PHCollectionMBS objects (asset collections or other collection lists) to be removed from the collection list.

This method removes child collections from the collection list based on their identity (determined by the localIdentifier property of each collection). To remove objects at specified indexes, use the removeChildCollections method.
See also:

- 12.16.12 removeChildCollections(AtIndexes as NSIndexSetMBS) 754

12.16.14 replaceChildCollections(AtIndexes as NSIndexSetMBS, ChildCollections() as PHCollectionMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Replaces the child collections at the specified indexes in the collection list with the specified collections.
Notes: indexes: The indexes of the child collections to be replaced in the collection list.

collections: An array of PHCollectionMBS objects (asset collections or other collection lists) to be inserted into (or moved within) the collection list.

To ensure that the index set you specify is valid even if the collection list has changed since you fetched it, create a change request with a snapshot of the collection list,Äôs contents using the changeRequestForCollectionList method before rearranging child collections.

12.16.15 Properties

12.16.16 placeholderForCreatedCollectionList as PHObjectPlaceholderMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A placeholder object for the collection list that the change request creates.
Notes: Use this property if you need to reference the collection created by a change request within the same change block. For details on change blocks, see PHPhotoLibraryMBS.
(Read only property)

12.16.17 title as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The displayed name of the collection list.
Notes: Set this property to change the collection list,Äôs title.
(Read and Write property)

12.17 class PHCollectionListMBS

12.17.1 class PHCollectionListMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A group containing Photos asset collections, such as Moments, Years, or folders of user-created albums.
Notes: In the Photos framework, collection objects (including asset collections) do not directly reference their member objects, and there are no other objects that directly reference collection objects. To retrieve the members of a collection list, fetch them with a PHCollectionMBS class method such as fetchCollectionsInCollectionList. To find objects at the root of the collection list hierarchy (such as album folders with no parent folders), use the fetchTopLevelUserCollectionsWithOptions method.

Accessing or modifying the Photos library requires explicit authorization from the user. The first time you call one of the methods listed in Fetching Collection Lists, Photos automatically prompts the user for authorization. (Alternatively, you can use the PHPhotoLibraryMBS requestAuthorization: method to prompt the user at a time of your choosing.)

Your app,Äôs Info.plist file must provide a value for the NSPhotoLibraryUsageDescription key that explains to the user why your app is requesting Photos access. Apps linked on or after iOS 10.0 will crash if this key is not present.

Like assets and asset collections, collection lists are immutable. To create, rename, or delete collection lists, or to add, remove, or rearrange members in a collection list, create a PHCollectionListChangeRequestMBS object within a photo library change block. For details on using change requests and change blocks to update the photo library, see PHPhotoLibraryMBS.
Subclass of the PHCollectionMBS class.

12.17.2 Methods

12.17.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.17.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.17.5 fetchCollectionListsContainingCollection(collection as PHCollectionMBS, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves collection lists that contain the specified collection.
Notes: collection: An asset collection or another collection list.
options: Options that specify a filter predicate and sort order for the fetched collection lists, or nil to use default options. For details, see PHFetchOptionsMBS class.

Returns a fetch result that contains the requested PHCollectionList objects, or an empty fetch result if no objects match the request.

Different kinds of collections have different containment possibilities. For example, an asset collection whose type is TypeAlbum may be contained in a folder, or have no containing collection list. A folder, in turn, may be contained in another folder. An asset collection whose type is TypeMoment is always contained by two collection lists: a moment cluster and a moment year.

12.17.6 fetchCollectionListsWithLocalIdentifiers(identifiers() as string, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves collection lists with the specified local-device-specific unique identifiers.
Notes: identifiers: An array of strings, each the localIdentifier string of a collection list.
options: Options that specify a filter predicate and sort order for the fetched collection lists, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHCollectionListMBS objects, or an empty fetch result if no objects match the request.

12.17.7 fetchCollectionListsWithType(collectionListType as Integer, subtype as Integer, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves collection lists of the specified type.
Notes: collectionListType: A type of collection list. See PHCollectionListType.
subtype: A subtype of collection list. See PHCollectionListSubtype.

options: Options that specify a filter predicate and sort order for the fetched collection lists, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHCollectionListMBS objects, or an empty fetch result if no objects match the request.

By default, the returned PHFetchResult object contains all collection lists with the specified type and subtype. To retrieve a more specific set of collection lists, provide a PHFetchOptionsMBS object containing a filter predicate.

12.17.8 localizedLocationNames as String()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The names of locations grouped by the collection (an array of strings).
Notes: For a collection list representing a group of moments, as seen in the Collections view in the Photos app, this property lists the location names associated with each moment in the group. For other types of collection list, this property,Äôs value is nil.

12.17.9 transientCollectionListWithCollections(collections() as PHCollectionMBS, title as String) as PHCollectionListMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a temporary collection list that contains the specified asset collections.
Notes: collections: An array of PHAssetCollectionMBS objects.
title: A name for the new temporary collection list.

Returns a new collection list.

Transient collection lists are not saved to local storage or iCloud and do not appear in the Photos application or other apps using the Photos framework. A transient collection can be useful if you,Äôve designed a UI for displaying the contents of a collection list and want to display an arbitrary set of collections.

12.17.10 transientCollectionListWithCollectionsFetchResult(fetchResult as PHFetchResultMBS, title as String) as PHCollectionListMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a temporary collection list containing the asset collections in the specified fetch result.

Notes: fetchResult: A fetch result that contains one or more PHAssetCollection objects.
title: A name for the new temporary collection list.

Returns a new collection list.

Transient collection lists are not saved to local storage or iCloud and do not appear in the Photos application or other apps using the Photos framework. A transient collection can be useful if you,Äôve designed a UI for displaying the contents of a collection list and want to display an arbitrary set of collections.

12.17.11 Properties

12.17.12 collectionListSubtype as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of asset collection grouping the collection list represents.
Notes: Use subtypes to make minor distinctions between collection lists of the same type, such as moment clusters and moment years. See Subtype constants.
(Read only property)

12.17.13 collectionListType as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of asset collection group that the collection list represents.
Notes: A collection list may represent an upper level of the Moments hierarchy shown in the Photos app, a folder that contains albums, or a smart folder synced from iPhoto.
(Read only property)

12.17.14 endDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The latest creation date among all assets in the collection list.
Notes: This property applies only to collection lists whose type is TypeMomentList. For other collection list types, this property,Äôs value is nil.
(Read only property)

12.17.15 endDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The latest creation date among all assets in the collection list.
Notes: This property applies only to collection lists whose type is TypeMomentList. For other collection list types, this property,Äôs value is nil.
(Read only property)

12.17.16 startDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The earliest creation date among all assets in the collection list.
Notes: This property applies only to collection lists whose type is TypeMomentList. For other collection list types, this property,Äôs value is nil.
(Read only property)

12.17.17 startDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop & iOS.

Function: The earliest creation date among all assets in the collection list.
Notes: This property applies only to collection lists whose type is TypeMomentList. For other collection list types, this property,Äôs value is nil.
(Read only property)

12.17.18 Constants

Sub Types

Constant	Value	Description
SubtypeAny	-1	Use this value to fetch collection lists of all possible subtypes.
SubtypeMomentListCluster	1	The collection list is a moment cluster, grouping several related moments.
SubtypeMomentListYear	2	The collection list is a moment year, grouping all moments from one or more calendar years.
SubtypeRegularFolder	100	The collection list is a folder containing albums or other folders.
SubtypeSmartFolderEvents	200	The collection list is a smart folder containing one or more Events synced from iPhoto.
SubtypeSmartFolderFaces	201	The collection list is a smart folder containing one or more Faces synced from iPhoto.

Types

Constant	Value	Description
TypeFolder	2	A folder containing asset collections of type PHAssetCollectionTypeAlbum or PHAssetCollectionTypeSmartAlbum.
TypeMomentList	1	A group of asset collections of type PHAssetCollectionTypeMoment. Moment lists include both moment clusters and moment years. Moment clusters appear as ‚ÄúCollections‚Äù in the Photos app, grouping individual moments. Years group all moments containing assets created in the same calendar year.
TypeSmartFolder	3	A smart folder synced to the device.

12.18 class PHCollectionMBS

12.18.1 class PHCollectionMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The abstract superclass for Photos asset collections and collection lists.
Notes: You do not create or work with instances of this class directly. Instead, use one of its two concrete subclasses, PHAssetCollectionMBS or PHCollectionListMBS.

- A PHAssetCollectionMBS object represents a collection of photo or video assets, such as an album, moment, or Shared Photo Stream.
- A PHCollectionListMBS object represents a collection that contains other collections, such as a a folder containing albums or the set of all moments in a calendar year.

Accessing or modifying the Photos library requires explicit authorization from the user. The first time you call one of the methods listed in Fetching Collections, Photos automatically prompts the user for authorization. (Alternatively, you can use the PHPhotoLibraryMBS requestAuthorization method to prompt the user at a time of your choosing.)

Your app,Äôs Info.plist file must provide a value for the NSPhotoLibraryUsageDescription key that explains to the user why your app is requesting Photos access. Apps linked on or after iOS 10.0 will crash if this key is not present.
Subclass of the PHObjectMBS class.

12.18.2 Methods

12.18.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.18.4 canPerformEditOperation(anOperation as Integer) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns whether the collection supports the specified editing operation.
Notes: anOperation: A bit mask of editing operations to be tested.

Returns true if the asset supports the specified editing operation; otherwise, false.

If an asset collection or collection list supports editing, you can create a PHAssetCollectionChangeRequestMBS or PHCollectionListChangeRequestMBS object inside a PHPhotoLibraryMBS change block to submit a change.

12.18.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.18.6 fetchCollectionsInCollectionList(collectionList as PHCollectionListMBS, options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves collections from the specified collection list.
Notes: collectionList: The collection list from which to fetch collections.
options: Options that specify a filter predicate and sort order for the fetched collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHCollectionMBS objects, or an empty fetch result if no objects match the request.

By default, the returned PHFetchResultMBS object contains all collections in the specified collection list. To retrieve a more specific set of assets, provide a PHFetchOptionsMBS object that contains a filter predicate.

12.18.7 fetchTopLevelUserCollections(options as PHFetchOptionsMBS = nil) as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves collections from the root of the photo library,Äôs hierarchy of user-created albums and folders.
Notes: options: Options that specify a filter predicate and sort order for the fetched collections, or nil to use default options. For details, see PHFetchOptionsMBS.

Returns a fetch result that contains the requested PHCollection objects, or an empty fetch result if no objects match the request.

12.18.8 Properties

12.18.9 canContainAssets as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the collection can contain assets.
Notes: If this value is true, the collection is a PHAssetCollection object; otherwise, false. For details on asset collections, see PHAssetCollectionMBS.
(Read only property)

12.18.10 canContainCollections as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the collection can contain other collections.
Notes: If this value is true, the collection is a PHCollectionList object; otherwise, false. For details on collection lists, see PHCollectionListMBS.
(Read only property)

12.18.11 localizedTitle as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The localized name of the collection.
Notes: This property,Äôs value can be either a user-supplied name for a user-created collection, or the localized name of a collection built into the Photos app.
(Read only property)

12.18.12 Constants

Edit Operations

Constant	Value	Description
EditOperationAddContent	3	The collection supports adding items that already exist elsewhere in the photo library.
EditOperationCreateContent	4	The collection supports creating new items.
EditOperationDelete	6	The collection itself can be deleted.
EditOperationDeleteContent	1	The collection supports deleting the items it contains.
EditOperationRearrangeContent	5	The collection supports reordering the arrangement of items it contains.
EditOperationRemoveContent	2	The collection supports removing the items it contains.
EditOperationRename	7	The collection itself can be renamed.

12.19 class PHContentEditingInputMBS

12.19.1 class PHContentEditingInputMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A container that provides information about and access to the image, video, or Live Photo content of an asset to be edited.
Notes: To edit an asset,Äôs photo or video content:

1. Fetch a PHAsset object that represents the photo or video to be edited.
2. Call the asset,Äôs requestContentEditingInputWithOptions method to retrieve a PHContentEditingInputMBS object.
3. Apply your edits to the asset. To allow a user to continue working with the edit later (for example, to adjust the parameters of a photo filter), create a PHAdjustmentDataMBS object describing the changes.
4. Initialize a PHContentEditingOutputMBS object. For photo- or video-only assets, use the editing output,Äôs properties to provide edited asset data. For Live Photo assets, create a PHLivePhotoEditingContextMBS object to edit the Live Photo content.
5. Use a photo library change block to commit the edit. In the block, create a PHAssetChangeRequestMBS object and set its contentEditingOutput property to the editing output that you created. For more details, see PHPhotoLibraryMBS.

You can also edit assets from photo editing extensions. In this case, instead of working with a PHAsset object, you implement methods in the PHContentEditingControllerMBS protocol. Photos provides a PHContentEditingInputMBS object when your extension begins editing. When editing is complete, Photos requests a PHContentEditingOutputMBS object that contains the edited asset content.

12.19.2 Methods

12.19.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.11 or newer.

12.19.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.19.5 Properties

12.19.6 adjustmentData as PHAdjustmentDataMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that describes the most recent edit to the asset,Äôs content.
Notes: Adjustment data describes the ‚Äúrecipe‚Äù for the last edit made to an asset,Äôs photo or video content. For example, a photo editing app can use this property to read information about a set of filters applied to a photo. With this information, your app can later allow a user to change the filter parameters.
(Read only property)

12.19.7 audiovisualAsset as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The video asset, as an AVAssetMBS object.
Notes: This object provides access to the video asset as a collection of tracks and metadata. For details on working with AVAssetMBS objects.

We pass result, the AVAssetMBS object, as variant to avoid plugin dependencies.
(Read only property)

12.19.8 creationDate as Date

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The date and time when the asset was originally created.
Notes: (Read only property)

12.19.9 creationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: The date and time when the asset was originally created.
Notes: (Read only property)

12.19.10 displaySizeImage as NSImageMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An image of the asset,Äôs contents, appropriately sized for display.
Notes: This property does not provide the full-sized image for a photo asset but rather a scaled-down image appropriate for use in a photo editing user interface. To load the full-sized asset image, use the fullSizeImageURL property.
(Read only property)

12.19.11 fullSizeImageOrientation as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The Exif display orientation of the full-size image file.
Notes: This property's value is a raw numeric value describing the encoded image orientation according to the TIFF and Exif specifications. To more easily work with such values, convert this value to the CGImagePropertyOrientation type.
(Read only property)

12.19.12 fullSizeImageURL as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The URL to a file that contains the full-sized image data.
Notes: Typically, your app or extension does not need to load a full-sized image for use in an editing UI. Instead, use the displaySizeImage property to retrieve an image suitable for screen display. You can then load the full-sized image on a background queue so that it will be ready by the time the user finishes editing the display-size image. At that time, apply the user,Äôs adjustments to the full-sized image and then use the PHContentEditingOutputMBS class to commit the edit to the photo library.
(Read only property)

12.19.13 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.19.14 livePhoto as PHLivePhotoMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The unedited Live Photo content of the editing input.
Notes: To edit the video and photo content of the Live Photo, create a PHLivePhotoEditingContextMBS

object using this PHContentEditingInputMBS object.
If the editing input does not represent a Live Photo, this property,Äôs value is nil, indicating that you cannot use this PHContentEditingInputMBS object to create a Live Photo editing context.
(Read only property)

12.19.15 location as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The location information that was saved with the asset.
Notes: Typically, an asset,Äôs location metadata identifies the place where the asset was captured.

We pass result, the CLLocationMBS object, as variant to avoid plugin dependencies.
(Read only property)

12.19.16 mediaSubtypes as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The subtypes of the asset, identifying special kinds of assets such as a panoramic photo or a high-frame-rate video.
Notes: See MediaSubtype* constants for possible values.
Because an asset can have more than one subtype, use these values as bit masks to identify an asset.
(Read only property)

12.19.17 mediaType as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of the asset, such as video or audio.
Notes: See MediaType* constants.
(Read only property)

12.19.18 playbackStyle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The playback style for the asset.
Notes: (Read only property)

12.19.19 uniformTypeIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The uniform type identifier for the asset,Äôs image or video data.
Notes: (Read only property)

12.19.20 Constants

Asset Playback Styles

Constant	Value	Description
PlaybackStyleImage	1	The asset should be displayed as a still image.
PlaybackStyleImageAnimated	2	The asset should be displayed as an animated image.
PlaybackStyleLivePhoto	3	The asset should be displayed as a Live Photo.
PlaybackStyleUnsupported	0	The asset has an unsupported or undefined media playback type.
PlaybackStyleVideo	4	The asset should be displayed as a video.
PlaybackStyleVideoLooping	5	The asset should be displayed as a looping video.

12.20 class PHContentEditingInputRequestOptionsMBS

12.20.1 class PHContentEditingInputRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the delivery of image or video data when you request to edit the content of a Photos asset.
Notes: You use the PHContentEditingInputRequestOptionsMBS class with the requestContentEditingInputWithOptions method for editing the contents of a PHAssetMBS object.
This class does not affect photo editing extensions.

12.20.2 Methods

12.20.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.20.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.20.5 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.20.6 SetCanHandleAdjustmentData(CanHandleAdjustmentData as CanHandleAdjustmentDataMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets delegate to be called when Photos needs to determine whether your app can continue previous edits made to an asset.

Notes: When an asset is edited, Photos stores a PHAdjustmentDataMBS object provided by the app or extension that edited the asset. This object provides all information necessary to reconstruct the edited asset using the original asset data. When your app requests to edit an asset, Photos calls this block to inquire whether your app can handle the asset,Äôs past adjustments.

12.20.7 SetProgressHandler(ProgressHandler as ProgressHandlerMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the delegate Photos calls periodically while downloading the asset.
Notes: If you request an asset whose data is not on the local device, and have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress and allow canceling the download.

12.20.8 Properties

12.20.9 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.20.10 NetworkAccessAllowed as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether Photos can download the requested asset from iCloud if needed.
Notes: If true (the default), Photos downloads the asset requested for editing if it is not stored on the local device. To be notified of the download,Äôs progress, use the PHContentEditingInputRequestOptionsMBS property to provide a delegate that Photos calls periodically while downloading the asset.
(Read and Write property)

12.20.11 ProgressHandler as ProgressHandlerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate Photos calls periodically while downloading the asset.
Notes: If you request an asset whose data is not on the local device, and have enabled downloading with the

networkAccessAllowed property, Photos calls your block periodically to report progress and allow canceling the download.
(Read and Write property)

12.20.12 Delegates

12.20.13 CanHandleAdjustmentDataMBS(adjustmentData as PHAdjustmentDataMBS, tag as Variant) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate takes the following parameter.
Notes: adjustmentData: A PHAdjustmentDataMBS object you can use to determine whether your app can work with past edits made to the asset. Typically, you make this decision based on the adjustment data,Äôs formatIdentifier and formatVersion properties.

If your method returns true, Photos provides the original asset data for editing. Your app uses the adjustment data to alter, add to, or reapply previous edits. (For example, an adjustment data may describe filters applied to a photo. Your app reapplies those filters and allows the user to change filter parameters, add new filters, or remove filters.)
If your method returns false, Photos provides the most recent asset data—the rendered output of all previous edits—for editing.

12.20.14 ProgressHandlerMBS(progress as double, byref stop as boolean, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate takes the following parameters.
Notes: progress: A floating-point value indicating the progress of the download. A value of 0.0 indicates the download has just started, and a value of 1.0 indicates the download is complete.
stop: Set stop to true inside the block to cancel the download.

12.21 class PHContentEditingOutputMBS

12.21.1 class PHContentEditingOutputMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A container to which you provide the results of editing the photo, video, or Live Photo content of a Photos asset.
Notes: To edit an asset,Äôs photo or video content:

1. Fetch a PHAssetMBS object that represents the photo or video to be edited.
2. Call the asset,Äôs requestContentEditingInputWithOptions method to retrieve a PHContentEditingInputMBS object. This object provides information about the asset, the asset data to be edited, and a preview image for display.
3. Apply your edits to the asset. To allow a user to continue working with the edit later (for example, to adjust the parameters of a photo filter), create a PHAdjustmentDataMBS object describing the changes.
4. Initialize a PHContentEditingOutput object. For photo- or video-only assets, provide the edited content with the renderedContentURL property. For Live Photo assets, create a PHLivePhotoEditingContext object to edit the Live Photo content and pass your content editing output to the saveLivePhotoToOutput method. For all asset types, provide your adjustment data with the adjustmentData property of the content editing output.
5. Use a photo library change block to commit the edit. (For details, see PHPhotoLibraryMBS.) In the block, create a PHAssetChangeRequestMBS object and set its contentEditingOutput property to the editing output that you created.
Each PHPhotoLibraryperformChanges call prompts the user for permission to edit the contents of the photo library—to edit multiple assets in one batch, create multiple PHAssetChangeRequest objects within the same change block, each with its own corresponding PHContentEditingOutput object.

You can also edit assets from photo editing extensions. In this case, instead of working with a PHAssetMBS object, you implement methods in the PHContentEditingController protocol. Photos provides a PHContentEditingOutputMBS object when your extension begins editing. When editing is complete, Photos requests a PHContentEditingOutputMBS object that contains the edited asset content.

Requires MacOS 10.11 or newer.

12.21.2 Methods

12.21.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.11 or newer.

12.21.4 Constructor(contentEditingInput as PHContentEditingInputMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an editing output from the specified editing input.
Notes: contentEditingInput: An object providing information about the asset to be edited.

Returns an initialized content editing output.

To complete the edit, use the renderedContentURL property to provide the edited asset content. Then, use the PHAssetChangeRequestMBS class or PHContentEditingController protocol to commit the edit to storage.
See also:

- 12.21.5 Constructor(placeholderForCreatedAsset as PHObjectPlaceholderMBS) 775

12.21.5 Constructor(placeholderForCreatedAsset as PHObjectPlaceholderMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an editing output for use in adding a new asset to the photo library.
Notes: placeholderForCreatedAsset: A placeholder object that ties the editing output to a PHAssetChangeRequest object for creating a new asset.

Returns an initialized content editing output.

Use this method if you want to add a new asset to the Photos library with edited content, as opposed to editing the content of an asset after adding it to the library. For example, you might use this option if your app applies filters to photos it captures with the device camera—instead of saving only the filtered image to the Photos library, your app can save both the filtered and the original image, allowing the user to revert to the original image or apply different filters later.
See also:

- 12.21.4 Constructor(contentEditingInput as PHContentEditingInputMBS) 775

12.21.6 Properties

12.21.7 adjustmentData as PHAdjustmentDataMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object describing the changes made to the asset.
Notes: You use adjustment data to describe the ‚Äúrecipe‚Äù for an edit that later edits can make use of. For example, a photo editing app can use this property to save information about the filters applied to a

photo. Later, the same app (or another app that understands its adjustment data format) can load the filter information, change the filter parameters, and reapply the filters to the original photo.

If you write new asset content to the URL specified by the renderedContentURL property, you must also provide a new, distinct PHAdjustmentDataMBS object describing your edit. Passing a preexisting adjustment data object (that describes an earlier edit) results in undefined behavior.
(Read only property)

12.21.8 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.21.9 renderedContentURL as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The URL at which to write a file containing edited asset content.
Notes: Read this property to find a URL for writing edited asset content. Then, if editing a photo asset, write the altered photo image to a file in JPEG format at this URL. If editing a video asset, export the video to a QuickTime (.mov) file at this URL.

Edited asset content must incorporate (or ,Äúbake in,Äù) the intended orientation of the asset. That is, the orientation metadata (if any) that you write in the output image or video file must declare the ,Äúup,Äù orientation, and the image or video data must appear right-side up when presented without orientation metadata.

For Live Photo content, you don,Äôt write output to this URL. Instead, pass the editing output object to the saveLivePhotoToOutput method.
(Read only property)

12.22 class PHFetchOptionsMBS

12.22.1 class PHFetchOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options that affect the filtering, sorting, and management of results that Photos returns when you fetch asset or collection objects.
Notes: Using class methods on the PHAssetMBS, PHCollectionMBS, PHAssetCollectionMBS, and PHCollectionListMBS classes to fetch assets or collections produces a PHFetchResultMBS object containing the requested objects. The options you specify control which objects the fetch result includes, how those objects are arranged in the fetch result, and how Photos should notify your app of changes to the fetch result.

Photos supports only a restricted set of keys for the predicate and sortDescriptors properties. The set of available keys depends on which class you,Äôre using to fetch assets or collections—see Table 1 for the list of keys supported by each class.

Supported predicate and sort descriptor keys

Class for Fetch Method	Supported Keys
PHAsset	SELF, localIdentifier, creationDate, modificationDate, mediaType, mediaSubtypes, duration, pixelWidth, pixelHeight, favorite (or isFavorite), hidden (or isHidden), burstIdentifier
PHAssetCollection	SELF, localIdentifier, localizedTitle (or title), startDate, endDate, estimatedAssetCount
PHCollectionList	SELF, localIdentifier, localizedTitle (or title), startDate, endDate
PHCollection (can fetch a mix of PHCollectionList and PHAssetCollection objects)	SELF, localIdentifier, localizedTitle (or title), startDate, endDate

Available in MacOS 10.13 or newer.

12.22.2 Methods

12.22.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.22.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.22.5 copy as PHFetchOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

12.22.6 setSortDescriptors(sortDescriptors() as NSSortDescriptorMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the list of sort descriptors, specifying an order for the fetched objects.

12.22.7 sortDescriptors as NSSortDescriptorMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A list of sort descriptors, specifying an order for the fetched objects.

12.22.8 Properties

12.22.9 fetchLimit as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum number of objects to include in the fetch result.
Notes: With the default fetch limit of zero, Photos returns all requested assets or collections in a fetch result. Change this value to fetch more efficiently in situations where a potentially very large result is not needed. For example, to fetch only the most recently captured asset, call the fetchAssetsWithOptions method, using the sortDescriptors property to sort in descending date order, and setting a fetch limit of one.
(Read and Write property)

12.22.10 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.22.11 includeAllBurstAssets as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the fetch result includes all assets from burst photo sequences.
Notes: When the user takes a sequence of photos in burst mode with the Camera app (on supported devices), the Photos app user interface groups the resulting assets together and allows the user to select favorite members of the sequence. Photos also automatically marks members of the sequence as potential user favorites.

If the value is false (the default), fetches that include burst photo sequences return only the user-picked members and representative asset of each sequence. If the value is true, such fetches include all assets in each sequence.
(Read and Write property)

12.22.12 includeAssetSourceTypes as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The set of source types for which to include assets in the fetch result.
Notes: Asset source types identify the means by which an asset enters the Photos library, and affect the possible actions you can perform on an asset. For example, assets synced from iTunes cannot be edited or deleted.

The SourceType type is an option set—to include multiple source types in the same query, combine type constants with the bitwise OR operator. See SourceType constants in PHAssetMBS class.
(Read and Write property)

12.22.13 includeHiddenAssets as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the fetch result includes assets marked as hidden.
Notes: If the value is false (the default), fetches exclude assets whose hidden property is true. If the value is true, fetches include all assets regardless of their hidden state.
(Read and Write property)

12.22.14 predicate as NSPredicateMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A predicate that specifies which properties to select results by and that also specifies any constraints on selection.
Notes: Construct a predicate with the properties of the class of objects that you want to fetch
(Read and Write property)

12.22.15 wantsIncrementalChangeDetails as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether your app receives detailed change information for the objects in the fetch result.
Notes: If you have registered a change observer with the shared PHPhotoLibraryMBS object, fetching assets or collections automatically registers your observer to receive information about later changes to the fetch result and about the objects it contains. For a fetch result, change information (a PHFetchResultChangeDetailsMBS object) can include a detailed list of incremental differences from the previous state of the fetch result, such as new photos captured since the original fetch.

If true (the default), Photos sends detailed incremental changes when such information is available. If false, Photos tells your app only when the fetch result has changed (in which case you can perform the fetch again to receive updated results).
(Read and Write property)

12.23 class PHFetchResultChangeDetailsMBS

12.23.1 class PHFetchResultChangeDetailsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A description of changes that occurred in the set of asset or collection objects listed in a fetch result.
Notes: A PHFetchResultChangeDetailsMBS object provides detailed information about the differences between two fetch results—one that you previously obtained and an updated one that would result if you performed the same fetch again. The change details object provides information useful for updating a UI that lists the contents of a fetch result, such as the indexes of added, removed, and rearranged objects.

12.23.2 Methods

12.23.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.23.4 changeDetailsFromFetchResult(fromResult as PHFetchResultMBS, toFetchResult as PHFetchResultMBS, changedObjects() as PHObjectMBS) as PHFetchResultChangeDetailsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a change details object that summarizes the differences between two fetch results.
Notes: fromResult: A fetch result to be treated as the ‚Äúbefore‚Äù state in the resulting change details object.
toResult: A fetch result to be treated as the ‚Äúafter‚Äù state in the resulting change details object.
changedObjects: An collection of objects to manually note as changed between the two fetch results.

Typically, you use the PHChangeMBS class to retrieve PHFetchResultChangeDetailsMBS objects describing any changes that have occurred since you performed a fetch, but you can also use this method to get a change details object that summarizes the difference between two arbitrary PHFetchResultMBS objects. In this case, Photos cannot automatically determine when the same objects are present in both fetch results but those objects‚Äô content has changed, so the changedIndexes and changedObjects properties are empty.

However, you can use the changedObjects parameter to manually note objects as changed. For example, consider a view controller that maintains a base fetch result representing a list of albums, and a transient

collection that filters the list according to the user,Äôs search terms.

12.23.5 changedObjects() as PHObjectMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The objects in the fetch result whose content or metadata have been updated.
Notes: An array of PHAssetMBS, PHAssetCollectionMBS, or PHCollectionListMBS objects.
If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.

12.23.6 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.23.7 enumerateMoves(theDelegate as enumerateMovesHandlerMBS, Tag as Variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Runs the specified block for each case where an object has moved from one index to another in the fetch result.
Notes: The toIndex parameter in the handler block is relative to the state of the fetch result after you,Äôve applied the changes described by the removedIndexes, insertedIndexes and changedIndexes properties. Therefore, if you use this method to update a collection view or similar user interface displaying the contents of the fetch result, update your UI to reflect insertions, removals, and changes before you process moves.

12.23.8 insertedObjects() as PHObjectMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The new items that have been inserted in the fetch result.
Notes: This array can contain PHAssetMBS, PHAssetCollectionMBS or PHCollectionListMBS objects, or some combination thereof.
If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.

12.23.9 removedObjects() as PHObjectMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The items that have been removed from the fetch result.
Notes: This array can contain PHAssetMBS, PHAssetCollectionMBS, or PHCollectionListMBS objects, or some combination thereof.
If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.

12.23.10 Properties

12.23.11 changedIndexes as NSIndexSetMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The indexes of objects in the fetch result whose content or metadata have been updated.
Notes: Use this index set to update elements in a collection view or similar user interface that displays the contents of the fetch result. These indexes are relative to the original fetch result (the fetchResultBeforeChanges property) after you,Äôve applied the changes described by the removedIndexes and insertedIndexes properties; when updating your app,Äôs interface, apply changes after removals and insertions and before moves.

If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.
(Read only property)

12.23.12 fetchResultAfterChanges as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The current fetch result, incorporating recent changes.
Notes: You can use this object to inspect the current state of the fetched objects even if the hasIncrementalChanges property,Äôs value is false. Using this fetch result is equivalent to performing once more the same fetch that returned the original fetch result.
(Read only property)

12.23.13 fetchResultBeforeChanges as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The original fetch result, without recent changes.
Notes: This property,Äôs value is the same object you passed to the changeDetailsForFetchResult method to request change details.

(Read only property)

12.23.14 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.23.15 hasIncrementalChanges as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether changes to the fetch result can be described incrementally.
Notes: If this value is true, use the insertedIndexes, removedIndexes, and changedIndexes properties (or the insertedObjects, removedObjects, and changedObjects properties) to find out which objects in the fetch result have been added, removed, or updated. You can also use the hasMoves property and enumerateMoves method to find out which objects in the fetch result have been rearranged. These properties can be useful for updating a collection view or similar interface that displays the fetch result,Äôs contents.

If this value is false, the fetch result is too different from its original state for incremental change information to be meaningful. Use the fetchResultAfterChanges property to get the fetch result,Äôs current membership. (If displaying the fetch result,Äôs contents, reload your user interface to match the new fetch result.)
(Read only property)

12.23.16 hasMoves as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether objects have been rearranged in the fetch result.
Notes: If this value is true, use the enumerateMoves method to find out which elements have been moved and what their new indexes are.
(Read only property)

12.23.17 insertedIndexes as NSIndexSetMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The indexes where new objects have been inserted in the fetch result.
Notes: Use this index set can to insert elements in a collection view or similar user interface that displays

the contents of the fetch result. These indexes are relative to the original fetch result (the fetchResultBeforeChanges property) after you,Äôve applied the changes described by the removedIndexes property; when updating your app,Äôs interface, apply insertions after removals and before changes and moves.

If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.
(Read only property)

12.23.18 removedIndexes as NSIndexSetMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The indexes from which objects have been removed from the fetch result.
Notes: Use this index set to remove elements from a collection view or similar user interface that displays the contents of the fetch result. These indexes are relative to the original fetch result (the fetchResultBeforeChanges property); when updating your app,Äôs interface, apply removals before insertions, changes, and moves.
If the hasIncrementalChanges property,Äôs value is false, this property,Äôs value is nil.
(Read only property)

12.23.19 Delegates

12.23.20 enumerateMovesHandlerMBS(fromIndex as Integer, toIndex as Integer, tag as variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls to provide details about which objects in the fetch result have moved to which indexes.
Notes: fromIndex: The index of an object in the original fetch result.
toIndex: The index to which the object has moved in the new fetch result.

12.24 class PHFetchResultMBS

12.24.1 class PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An ordered list of assets or collections returned from a Photos fetch method.
Notes: When you use class methods on the PHAssetMBS, PHCollectionMBS, PHAssetCollectionMBS, and PHCollectionListMBS classes to retrieve objects, Photos provides the resulting objects in a fetch result. You access the contents of a fetch result with various methods. Unlike an array, however, a PHFetchResult object dynamically loads its contents from the Photos library as needed, providing optimal performance even when handling a large number of results.

A fetch result provides thread-safe access to its contents. After a fetch, the fetch result,Äôs count value is constant, and all objects in the fetch result keep the same localIdentifier value. (To get updated content for a fetch, register a change observer with the shared PHPhotoLibraryMBS object.)

A fetch result caches its contents, keeping a batch of objects around the most recently accessed index. Because objects outside of the batch are no longer cached, accessing these objects results in refetching those objects. This process can result in changes to values previously read from those objects.
Blog Entries

- Show Live Photos in your Xojo application

12.24.2 Methods

12.24.3 allObjects as Variant()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Queries all objects as array.

12.24.4 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.24.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.24.6 containsObject(anObject as Variant) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns whether the specified object is present in the fetch result.
Notes: anObject: An object.

Returns true if anObject is present in the fetch result, otherwise false.

This method determines whether anObject is present in the fetch result by sending an isEqual: message to each of the fetch result,Äôs objects (and passing anObject as the parameter to each isEqual message).

12.24.7 copy as PHFetchResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the result set.

12.24.8 countOfAssetsWithMediaType(MediaType as Integer) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the number of assets in the fetch result of a specified type.
Notes: mediaType: The type of assets to count, such as image or video. See MediaType constants.

Returns the number of assets in the fetch result of the specified type.

The first time you call this method, Photos enumerates the contents of the fetch result to count those of the specified type, then caches the result. Subsequent calls with the same mediaType parameter return the cached value.
This method counts only the PHAssetMBS objects in a fetch result. If a fetch result contains only PHAssetCollectionMBS or PHCollectionListMBS objects, the return value is 0.

12.24.9 indexOfObject(anObject as Variant) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the lowest index whose corresponding object in the fetch result is equal to the specified object.
Notes: anObject: An object.

Returns the lowest index whose corresponding object in the fetch result is equal to anObject, or NSNotFound if no such object is in the fetch result.

Starting at index 0, this method sends an isEqual message to each object in the fetch result until it finds a match or reaches the end of the fetch result. This method passes the anObject parameter to each isEqual message.
See also:

- 12.24.10 indexOfObject(anObject as Variant, range as NSRangeMBS) as Integer 788

12.24.10 indexOfObject(anObject as Variant, range as NSRangeMBS) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the lowest index within the specified range whose corresponding object in the fetch result is equal to the specified object.
Notes: anObject: An object.
range: The range of indexes in the fetch result within which to search for anObject.

Returns the lowest index within range whose corresponding object in the fetch result is equal to anObject, or NSNotFound if no such object is in the fetch result.

Starting at range.location, this method sends an isEqual message to each object in the fetch result until it finds a match or reaches the end of the fetch result. This method passes the anObject parameter to each isEqual message.
Raises an exception (NSExceptionMBS class) if the range parameter represents a range that doesn,Äôt exist in the fetch result.
See also:

- 12.24.9 indexOfObject(anObject as Variant) as Integer 787

12.24.11 objectAtIndex(index as Integer) as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the object located at the specified index.
Notes: index: An index within the bounds of the fetch result.

Returns the object located at index in the fetch result.

Raises an exception (NSExceptionMBS class) if index is beyond the end of the fetch result (that is, greater than or equal to the value of the count property).

12.24.12 objectsAtIndexes(indexes as NSIndexSetMBS) as Variant()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an array containing the objects in the fetch result at the indexes in the specified index set.
Notes: indexes: An index set containing indexes within the bounds of the fetch result.

Returns an array containing the objects in the fetch result at the indexes specified by indexes.

The ordering of the returned array follows the index set. That is, in the returned array, an object with a higher index in the index set comes after any object with a smaller index in the index set.
Raises an exception (NSExceptionMBS) if any index in the index set is beyond the end of the fetch result (that is, greater than or equal to the value of the count property).

12.24.13 Properties

12.24.14 count as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The number of objects in the fetch result.
Notes: (Read only property)

12.24.15 firstObject as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The first object in the fetch result.
Notes: You specify the ordering of a fetch result in the PHFetchOptionsMBS object you pass to a fetch method.
Returns nil if the fetch result is empty.
(Read only property)

12.24.16 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.24.17 lastObject as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The last object in the fetch result.
Notes: You specify the ordering of a fetch result in the PHFetchOptionsMBS object you pass to a fetch method.
Returns nil if the fetch result is empty.
(Read only property)

12.24.18 Constants

Edit Operation

Constant	Value	Description
EditOperationContent	2	The asset,Äôs photo or video content can be edited. To begin the process of editing an asset, use the requestContentEditingInputWithOptions method.
EditOperationDelete	1	The asset can be deleted from the photo library. To delete one or more assets, create a change request with the deleteAssets method inside a PHPhotoLibrary change block.
EditOperationProperties	3	The asset,Äôs metadata properties can be edited. To change an asset,Äôs properties, create a change request with the changeRequestForAsset method inside a PHPhotoLibraryMBS change block.

Media Types

Constant	Value	Description
MediaTypeAudio	3	The asset is an audio file.
MediaTypeImage	1	The asset is a photo or other static image.
MediaTypeUnknown	0	The asset,Äôs type is unknown.
MediaTypeVideo	2	The asset is a video file.

Asset Playback Styles

Constant	Value	Description
PlaybackStyleImage	1	The asset should be displayed as a still image.
PlaybackStyleImageAnimated	2	The asset should be displayed as an animated image.
PlaybackStyleLivePhoto	3	The asset should be displayed as a Live Photo.
PlaybackStyleUnsupported	0	The asset has an unsupported or undefined media playback type.
PlaybackStyleVideo	4	The asset should be displayed as a video.
PlaybackStyleVideoLooping	5	The asset should be displayed as a looping video.

12.25 class PHImageManagerMBS

12.25.1 class PHImageManagerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that facilitates retrieving or generating preview thumbnails and asset data.
Notes: Use these methods to fetch full-size photo assets or thumbnail images, or to retrieve AVFoundation objects for playing, exporting, and manipulating video assets.

To load image or video data:
1. Use the PHAsset class to fetch the asset you,Äôre interested in.
2. Call the defaultManager method to retrieve the shared image manager object.
3. Use one of the methods listed in Requesting Images and Requesting Video Objects to load the asset,Äôs image or video data.

The image manager caches the asset images and data it provides, so later requests for the same assets with similar parameters will return results more quickly.

If you need to load image data for many assets together, use the PHCachingImageManagerMBS class to ,Äúpreheat,Äù the cache by loading images you expect to need soon. For example, when populating a collection view with photo asset thumbnails, you can cache images ahead of the current scroll position.

Available in MacOS 10.13 or later.
Blog Entries

- Show Live Photos in your Xojo application

12.25.2 Methods

12.25.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.25.4 cancelImageRequest(requestID as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels an asynchronous request.

Notes: requestID: The numeric identifier of the request to be canceled.

When you perform an asynchronous request for image data using the requestImageForAsset method, or for a video object using one of the methods listed in Requesting Video Objects, the image manager returns a numeric identifier for the request. To cancel the request before it completes, provide this identifier when calling the cancelImageRequest method.

12.25.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.25.6 defaultManager as PHImageManagerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the shared image manager object.
Notes: This method always returns the same image manager object, which is shared for all uses in your app.

12.25.7 MaximumSize as CGSizeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A special value for requesting original image data or the largest rendered image available.
Notes: When you use the PHImageManagerMaximumSize option, Photos provides the largest image available for the asset without scaling or cropping. (That is, it ignores the resizeMode option.)

12.25.8 PHImageCancelledKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary.
Notes: A key whose value indicates whether the image request was canceled.

This key provides information about an image loading result in the resultHandler block for methods listed in Requesting Images. The corresponding value is a Boolean value.If you call the cancelImageRequest method to cancel a request, Photos calls your result handler block with the value true for this key.

12.25.9 PHImageErrorKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary.
Notes: A key whose value is an error that occurred when Photos attempted to load the image.

This key provides information about an image loading result in the resultHandler block for methods listed in Requesting Images. The corresponding value is an NSErrorMBS object.Photos provides an error object for this key if it cannot provide an image for your handler block,Äôs result parameter. Examine the error object for information about the cause of the error.

12.25.10 PHImageResultIsDegradedKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary.
Notes: A key whose value indicates whether the result image is a low-quality substitute for the requested image.

This key provides information about an image loading result in the resultHandler block for methods listed in Requesting Images. The corresponding value is a Boolean value. If true, the result parameter of your resultHandler block contains a low-quality image because Photos could not yet provide a higher-quality image. Depending on your settings in the PHImageRequestOptions object that you provided with the request, Photos may call your result handler delegate again to provide a higher-quality image.

12.25.11 PHImageResultIsInCloudKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary.
Notes: A key whose value indicates whether photo asset data is stored on the local device or must be downloaded from iCloud.

This key provides information about an image loading result in the resultHandler delegate for methods listed in Requesting Images. The corresponding value is a Boolean value.

If true, no image was provided, because the asset data must be downloaded from iCloud. To download the data, submit another request, and specify true for the networkAccessAllowed option.

12.25.12 PHImageResultRequestIDKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary.
Notes: A key whose value is a unique identifier for the image request.

This key provides information about an image loading result in the resultHandler block for methods listed in Requesting Images. The corresponding value is an integer value. This identifier matches that returned when making a request. You can use it with the cancelImageRequest method to cancel requests with pending results that are no longer needed.

12.25.13 RequestAVAssetForVideo(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, exportPreset as String, CompletionHandler as RequestAVAssetForVideoCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests AVFoundation objects representing the video asset,Äôs content and state, to be loaded asynchronously.
Notes: asset: The video asset for which video objects are to be loaded.
options: Options specifying how Photos should handle the request and notify your app of progress or errors. For details, see PHVideoRequestOptionsMBS class.

Returns a numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

When you call this method, Photos downloads the video data (if necessary) and creates AVFoundation objects. It then calls your resultHandler block to provide the requested video.

Use this method when you want to work with the arrangement of audio and video tracks that an asset contains. If you plan to use the asset only for playback, call the requestPlayerItemForVideo method. If you plan to export the asset data, call the requestExportSessionForVideo method.

12.25.14 RequestExportSessionForVideo(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, exportPreset as String, CompletionHandler as RequestExportSessionForVideoCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests an export session for writing the video asset,Äôs data to a file, to be loaded asynchronously.
Notes: asset: The video asset for which an export session is to be created.
options: Options specifying how Photos should handle the request and notify your app of progress or errors. For details, see PHVideoRequestOptionsMBS.
exportPreset: The export preset name for exporting the asset. For available presets, see AVAssetExportSessionMBS.

A numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

When you call this method, Photos downloads the video data (if necessary) and creates an export session. It then calls your resultHandler delegate to provide the requested video.
For additional export options, use the requestAVAssetForVideo method and then create AVAssetReader and AVAssetWriterMBS objects to transcode and output the video asset,Äôs data.

12.25.15 RequestImageDataAndOrientationForAsset(asset as PHAssetMBS, options as PHImageRequestOptionsMBS, CompletionHandler as RequestImageForAssetCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests the largest represented image as data bytes and EXIF orientation for the specified asset.
Notes: asset: The asset for which to load image data.
options: Options specifying how Photos should handle the request, format the requested image, and notify your app of progress or errors.

If PHImageRequestOptionsMBS.VersionCurrent is requested and the asset has adjustments, the largest rendered image data is returned. In all other cases, the original image data is returned.
For further details, see PHImageRequestOptionsMBS class.

Returns a numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

When you call this method, Photos loads the largest available representation of the image asset, then calls your resultHandler block to provide the requested data. Depending on the options you specify and the current state of the asset, Photos may download asset data from the network.

By default, this method executes asynchronously. If you call it from a background thread, you may change the synchronous property of the options parameter to true to block the calling thread until either the requested image is ready or an error occurs, at which time Photos calls your result handler. This method

ignores the deliveryMode option—Photos calls your result handler block exactly once.

If the version option is set to PHImageRequestOptionsMBS.VersionCurrent, Photos provides rendered image data, including the results of any edits that have been made to the asset content. Otherwise, Photos provides the originally captured image data for the asset.

12.25.16 RequestImageForAsset(asset as PHAssetMBS, targetSize as CGSizeMBS, contentMode as Integer, options as PHImageRequestOptionsMBS, CompletionHandler as RequestImageForAssetCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests an image representation for the specified asset.
Notes: asset: The asset whose image data is to be loaded.
targetSize: The target size of image to be returned.
contentMode: An option for how to fit the image to the aspect ratio of the requested size. For details, see PHImageContentMode.
options: Options specifying how Photos should handle the request, format the requested image, and notify your app of progress or errors. For details, see PHImageRequestOptionsMBS class.

Returns a numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

When you call this method, Photos loads or generates an image of the asset at, or near, the size you specify. Next, it calls your resultHandler block to provide the requested image. To serve your request more quickly, Photos may provide an image that is slightly larger than the target size—either because such an image is already cached or because it can be generated more efficiently. Depending on the options you specify and the current state of the asset, Photos may download asset data from the network.

By default, this method executes asynchronously. If you call it from a background thread you may change the synchronous property of the options parameter to true to block the calling thread until either the requested image is ready or an error occurs, at which time Photos calls your result handler.

For an asynchronous request, Photos may call your result handler block more than once. Photos first calls the block to provide a low-quality image suitable for displaying temporarily while it prepares a high-quality image. (If low-quality image data is immediately available, the first call may occur before the method returns.) When the high-quality image is ready, Photos calls your result handler again to provide it. If the image manager has already cached the requested image at full quality, Photos calls your result handler only once. The PHImageResultIsDegradedKey key in the result handler,Äôs info parameter indicates when Photos is providing a temporary low-quality image.

You can use this method for both photo and video assets —for a video asset, an image request provides a thumbnail image or poster frame.

12.25.17 RequestImageForAssetSync(asset as PHAssetMBS, targetSize as CGSizeMBS, contentMode as Integer, options as PHImageRequestOptionsMBS, byref Info as Dictionary) as NSImageMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests an image representation for the specified asset synchronously.
Notes: asset: The asset whose image data is to be loaded.
targetSize: The target size of image to be returned.
contentMode: An option for how to fit the image to the aspect ratio of the requested size. For details, see PHImageContentMode.
options: Options specifying how Photos should handle the request, format the requested image, and notify your app of progress or errors. For details, see PHImageRequestOptionsMBS class.

Returns the image.

12.25.18 RequestLivePhotoForAsset(asset as PHAssetMBS, targetSize as CGSizeMBS, contentMode as Integer, options as PHLivePhotoRequestOptionsMBS, CompletionHandler as RequestLivePhotoForAssetCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests a Live Photo representation for the specified asset.
Notes: asset: The asset whose Live Photo data is to be loaded.
targetSize: The target size of Live Photo to be returned.
contentMode: An option for how to fit the image to the aspect ratio of the requested size. For details, see PHImageContentMode.
options: Options specifying how Photos should handle the request, format the requested image, and notify your app of progress or errors. For details, see PHLivePhotoRequestOptionsMBS class.

Returns a numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

A Live Photo is a picture, taken with a supported device, that includes movement and sound from the moments just before and after its capture. Much like how a UIImage object represents a ready-to-use form of an image, a PHLivePhotoMBS object represents a Live Photo whose image, motion, and sound data are prepared for display. Use this method to request an asset,Äôs Live Photo form; after Photos calls your resultHandler delegate to provide the Live Photo, you can display it using the PHLivePhotoViewMBS class.

Use this method only when you plan to display the motion and sound content associated with a Live Photo. In contexts where you need only a still image for a Live Photo asset—for example, when loading thumbnails to display in a photo chooser interface—use the requestImageForAsset method instead.

When you call this method, Photos loads or generates a PHLivePhotoMBS object for the asset at, or near, the size you specify. Next, it calls your resultHandler block to provide the requested image. To serve your request more quickly, Photos may provide an image that is slightly larger than the target size—either because such an image is already cached or because it can be generated more efficiently. Depending on the options you specify and the current state of the asset, Photos may download asset data from the network. This method always executes asynchronously.

Photos may call your result handler block more than once. Photos first calls the block to provide a low-quality image suitable for displaying temporarily while it prepares a high-quality image. (If low-quality image data is immediately available, the first call may occur before the method returns.) When the high-quality image is ready, Photos calls your result handler again to provide it. If the image manager has already cached the requested image at full quality, Photos calls your result handler only once. The PHImageResultIsDegraded-Key key in the result handler,Äôs info parameter indicates when Photos is providing a temporary low-quality image.

12.25.19 RequestPlayerItemForVideo(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, CompletionHandler as RequestPlayerItemForVideoCompletedMBS, tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests a representation of the video asset for playback, to be loaded asynchronously.
Notes: asset: The video asset to be played back.
options: Options specifying how Photos should handle the request and notify your app of progress or errors. For details, see PHVideoRequestOptionsMBS.

A numeric identifier for the request. If you need to cancel the request before it completes, pass this identifier to the cancelImageRequest method.

When you call this method, Photos downloads the video data (if necessary) and creates a player item. It then calls your resultHandler block to provide the requested video.

Use this method when you want to simply play back the video asset as it currently exists. For more detailed options or to work with the asset,Äôs audio and video tracks, use the requestAVAssetForVideo method instead.

12.25.20 Properties

12.25.21 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.25.22 Constants

Constants

Constant	Value	Description
PHInvalidImageRequestID	0	A special value provided for asynchronous image requests that cannot be canceled.

Content Modes

Constant	Value	Description
ContentModeAspectFill	1	Scales the image so that it completely fills the target size.
ContentModeAspectFit	0	Scales the image so that its larger dimension fits the target size.
ContentModeDefault	0	Fits the image to the requested size using the default option, PHImageContentModeAspectFit. Use this content mode when requesting a full-sized image using the PHImageManagerMBS.MaximumSize value for the target size. In this case, the image manager does not scale or crop the image.

12.25.23 Delegates

12.25.24 RequestAVAssetForVideoCompletedMBS(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, AVAsset as Variant, AVAudioMix as Variant, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls after loading the asset,Äôs data.
Notes: asset: An object that provides access to the video asset as a collection of tracks and metadata. For details on working with AVAsset objects, see AVFoundation Programming Guide.
audioMix: Use this object to rearrange the asset,Äôs audio tracks, edit additional audio into the mix, or configure an AVAssetReaderOutput object for exporting the asset,Äôs audio data. If nil, the asset uses a default audio mix.
info: A dictionary providing information about the status of the request. See Image Result Info Keys for possible keys and values.

We pass AVAsset, the AVAssetMBS object, as variant to avoid plugin dependencies.
Same for AVAudioMix with AVAudioMixMBS object.

12.25.25 RequestExportSessionForVideoCompletedMBS(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, exportPreset as string, exportSession as Variant, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls after loading the asset,Äôs data and preparing the export session.
Notes: exportSession: An AVAssetExportSessionMBS object that you can use for writing the video asset,Äôs data to a file.
info: A dictionary that provides information about the status of the request. See Image Result Info Keys for possible keys and values.

We pass AVAsset, the AVAssetExportSessionMBS object, as variant to avoid plugin dependencies.

12.25.26 RequestImageDataAndOrientationForAssetCompletedMBS(asset as PHAssetMBS, options as PHImageRequestOptionsMBS, imageData as MemoryBlock, dataUTI as String, orientation as Integer, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate called, exactly once, when image loading is complete, providing the requested image or information about the status of the request.
Notes: imageData: The requested image.
dataUTI: The uniform type identifier for the image.
orientation: The EXIF orientation for the image, as a CGImagePropertyOrientation.
info: A dictionary providing information about the status of the request.

12.25.27 RequestImageForAssetCompletedMBS(asset as PHAssetMBS, options as PHImageRequestOptionsMBS, result as NSImageMBS, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate to be called when image loading is complete, providing the requested image or information about the status of the request.
Notes: result: The requested image.

info: A dictionary providing information about the status of the request. See Image Result Info Keys for possible keys and values.

12.25.28 RequestLivePhotoForAssetCompletedMBS(asset as PHAssetMBS, options as PHImageRequestOptionsMBS, livePhoto as PHLivePhotoMBS, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate to be called when image loading is complete, providing the requested image or information about the status of the request.
Notes: result: The requested Live Photo object.
info: A dictionary providing information about the status of the request. See Image Result Info Keys for possible keys and values.

12.25.29 RequestPlayerItemForVideoCompletedMBS(asset as PHAssetMBS, options as PHVideoRequestOptionsMBS, playerItem as Variant, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate Photos calls after loading the asset,Äôs data and preparing the player item.
Notes: playerItem: An AVPlayerItemMBS object that you can use for playing back the video asset.
info: A dictionary providing information about the status of the request. See Image Result Info Keys for possible keys and values.

We pass playerItem, the AVPlayerItemMBS object, as variant to avoid plugin dependencies.

12.26 class PHImageRequestOptionsMBS

12.26.1 class PHImageRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the delivery of still image representations of Photos assets you request from an image manager.
Notes: Available on macOS 10.13 or newer.

12.26.2 Methods

12.26.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.26.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.26.5 copy as PHImageRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the request.

12.26.6 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.26.7 SetProgressHandler(ProgressHandler as ProgressHandlerMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the delegate that Photos calls periodically while downloading the image.
Notes: If you request an image whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your delegate periodically to report progress and to allow you to cancel the download.

12.26.8 Properties

12.26.9 DeliveryMode as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The requested image quality and delivery priority.
Notes: Use this property to tell Photos to provide an image quickly (possibly sacrificing image quality), to provide a high-quality image (possibly sacrificing speed), or to provide both automatically if needed. See DeliveryMode constants.
(Read and Write property)

12.26.10 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.26.11 NetworkAccessAllowed as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether Photos can download the requested image from iCloud.
Notes: If true, and the requested image is not stored on the local device, Photos downloads the image from iCloud. To be notified of the download,Äôs progress, use the progressHandler property to provide a delegate that Photos calls periodically while downloading the image. If false (the default), and the image is not on the local device, the PHImageResultIsInCloudKey value in the result handler,Äôs info dictionary indicates that the image is not available unless you enable network access.
(Read and Write property)

12.26.12 normalizedCropRect as CGRectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A rectangle for requesting a cropped version of the original image.
Notes: To request a cropped image, specify the crop rectangle in a unit coordinate space relative to the image. In this coordinate system, the point { 0.0,0.0 } refers to the upper left corner of the image, and the point { 1.0,1.0 } refers to the opposite corner regardless of the image,Äôs aspect ratio.
This property defaults to zero rectangle, which specifies no cropping.
If you specify a crop rectangle, you must also specify the PHImageRequestOptionsResizeModeExact option for the resizeMode property.
(Read and Write property)

12.26.13 ProgressHandler as ProgressHandlerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls periodically while downloading the image.
Notes: If you request an image whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your delegate periodically to report progress and to allow you to cancel the download.
(Read and Write property)

12.26.14 ResizeMode as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mode that specifies how to resize the requested image.
Notes: Use this property to choose how to fit the image to the target size you specified when requesting image data. See PHImageRequestOptionsResizeMode.
(Read and Write property)

12.26.15 Synchronous as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether Photos processes the image request synchronously.
Notes: If false (the default), the requestImageForAsset method returns immediately. Depending on the deliveryMode property, Photos may call your resultHandler delegate before the method returns, at some later time, or both.
If true, the requestImageForAsset method blocks the calling thread until image data is ready or an error occurs. Photos calls your result handler delegate exactly once.

Perform synchronous requests from a background thread only.
(Read and Write property)

12.26.16 Version as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The version of the image to be requested.
Notes: Use this property to request a version of the image with or without adjustments, or to request high-quality original data (for example, a RAW file) if such is available. See Version constants.
(Read and Write property)

12.26.17 Constants

Delivery Modes

Constant	Value	Description
DeliveryModeFastFormat	2	Photos provides only a fast-loading image, possibly sacrificing image quality.
DeliveryModeHighQualityFormat	1	Photos provides only the highest-quality image available, regardless of how much time it takes to load.
DeliveryModeOpportunistic	0	Photos automatically provides one or more results in order to balance image quality and responsiveness.

Resize Mode

Constant	Value	Description
ResizeModeExact	2	Photos resizes the image to match the target size exactly. Resizing to exactly match a target size is less efficient than using the fast resizing option. You must choose this option if you use the normalizedCropRect property to request a cropped image.
ResizeModeFast	1	Photos efficiently resizes the image to a size similar to, or slightly larger than, the target size. With this option, Photos can use image subsampling to quickly provide an image at a size roughly matching the target size.
ResizeModeNone	0	Photos does not resize the image asset.

Version Modes

Constant	Value	Description
VersionCurrent	0	Request the most recent version of the image asset (the one that reflects all edits).
VersionOriginal	2	Request the original, highest-fidelity version of the image asset.
VersionUnadjusted	1	Request a version of the image asset without adjustments.

12.26.18 Delegates

12.26.19 ProgressHandlerMBS(progress as double, error as NSErrorMBS, byref stop as boolean, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls while downloading asset data from iCloud. Used by the progressHandler property.
Notes: If you request an image whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress and to allow you to cancel the download.

Photos calls this block in an arbitrary serial queue. Dispatch to the main thread if your handler needs to update the user interface.

The method takes the following parameters:
progress: A floating-point value indicating the progress of the download. A value of 0.0 indicates that the download has just started, and a value of 1.0 indicates the download is complete.
error: An NSErrorMBS object describing an error that occurred when attempting to download the image, or nil if no errors have occurred.
stop: A pointer to a Boolean value. To cancel the download, set stop to true inside the block.
info: A dictionary providing additional information about the status of the image request. See Image Result Info Keys for possible keys and values. For example, a true value for the key PHImageResultIsDegradedKey indicates that Photos may send the full-quality version of the image later, depending on the value of the image request,Äôs deliveryMode property.

If the method call is asynchronously dispatched to main thread, the stop property does not work.

12.27 control PHLivePhotoControlMBS

12.27.1 control PHLivePhotoControlMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A view that displays a Live Photo—a picture that also includes motion and sound from the moments just before and after its capture.
Notes: Use a Live Photo view to display the photo and control playback of its motion and sound content.

In macOS, Live Photo objects are available only when editing Live Photo content in a photo editing extension that runs in the Photos app—see the PHContentEditingInputMBS class to access Live Photo content in an editing session.

By default, a Live Photo view uses its own gesture recognizer to allow the user to play the motion and sound content of a Live Photo with the same interactions and visual effects seen in the Photos app. To customize this gesture recognizer—for example, to install it on a different view for proper event handling in your app,Äôs view hierarchy—use the playbackGestureRecognizer property.

To animate the view briefly to hint that a picture is a Live Photo, use the startPlayback method with the PlaybackStyleHint option.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- New in the MBS Xojo Plugins Version 20.2
- MBS Xojo Plugins, version 20.2pr1

Xojo Developer Magazine

- 18.3, page 57: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

12.27.2 Methods

12.27.3 available as boolean

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

12.27.4 startPlayback(style as integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Begins playback of Live Photo content in the view.
Notes: style: An option for how much of the Live Photo,Äôs motion and sound content to play. See PHLivePhotoViewPlaybackStyle.

Use the style parameter to choose whether to play the full motion and sound content of the Live Photo or only a brief section.

Typically, an app does not need to directly control playback, because a Live Photo view provides interactive playback control. Use this method only when non-interactive playback is appropriate—for example, to briefly animate the content to indicate that a view contains a Live Photo rather than a still image.

12.27.5 stopPlayback

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Ends playback of Live Photo content in the view.
See also:

- 12.27.6 stopPlayback(animated as boolean) 809

12.27.6 stopPlayback(animated as boolean)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Ends playback of Live Photo content in the view.
See also:

- 12.27.5 stopPlayback 809

12.27.7 Properties

12.27.8 audioVolume as Single

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The audio gain to apply to the Live Photo,Äôs movie content during playback.
Notes: Values for this property must be between 0.0 and 1.0, inclusive. A value of 1.0 (the default) plays audio content from the Live Photo at full volume (relative to the system volume). A value of 0.0 is equivalent to setting the muted property to true.

(Read and Write property)

12.27.9 ContentMode as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The content mode.
Notes: (Read and Write property)

12.27.10 LivePhoto as PHLivePhotoMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The Live Photo displayed in the view.
Notes: (Read and Write property)

12.27.11 livePhotoBadgeView as NSViewMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A view for displaying Live Photo status.
Notes: The PHLivePhotoViewMBS uses this subview to display icons indicating the existence or status of Live Photo content. Photos manages the content displayed in the badge view, so you don,Äôt need to do anything with view,Äôs content—instead, this property provides access to the badge view so you can change where it appears in your view hierarchy if needed. For example, if you display a Live Photo view within a scroll view, you can move the badge view so that its position remains constant while the scroll view scrolls.
(Read only property)

12.27.12 muted as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines whether the view plays the audio content of its Live Photo.
Notes: The default value is false, indicating that the view plays audio content along with the motion content of its Live Photo. Change this value to true to play motion content but not audio content.
(Read and Write property)

12.27.13 View as NSViewMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.
Notes: Use this object to set more options on the control.
(Read only property)

12.27.14 Events

12.27.15 BoundsChanged

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

12.27.16 Close

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

12.27.17 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

12.27.18 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Called when a menuitem is choosen.
Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

12.27.19 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

12.27.20 didEndPlaybackWithStyle(playbackStyle as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Notifies the delegate that Live Photo playback has ended.
Notes: playbackStyle: The style of playback, indicating whether the content was played in full or briefly previewed.

12.27.21 EnableMenuItems

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

12.27.22 FrameChanged

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

12.27.23 GotFocus

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

The control itself got focus.
In Xojo version 2021r3 and newer this event is named FocusReceived.
Notes:

This only fires if the control itself got focus and not a sub control.

12.27.24 LostFocus

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

The control lost focus.
In Xojo version 2021r3 and newer this event is named FocusLost.
Notes:

This only fires if the control itself lost focus and not a sub control.

12.27.25 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

12.27.26 MouseDrag(x as Integer, y as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

12.27.27 MouseUp(x As Integer, y As Integer)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

12.27.28 Open

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

The control is about to was created and you can initialize it.
In Xojo version 2021r3 and newer this event is named Opening.

12.27.29 ScaleFactorChanged(NewFactor as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

12.27.30 willBeginPlaybackWithStyle(playbackStyle as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Notifies the control that Live Photo playback is about to begin.
Notes: playbackStyle: The style of playback, indicating whether the content is to be played in full or briefly previewed.

12.27.31 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

12.27.32 Constants

Content Modes

Constant	Value	Description
ContentModeAspectFill	1	Aspect Fill.
ContentModeAspectFit	0	Aspect Fit.

Playback Styles

Constant	Value	Description
PlaybackStyleFull	1	Plays back the entire motion and sound content of the Live Photo, including transition effects at the start and end.
PlaybackStyleHint	2	Plays back only a brief section of the motion content of the Live Photo, without sound.
PlaybackStyleUndefined	0	This value is invalid for use.

12.28 class PHLivePhotoEditingContextMBS

12.28.1 class PHLivePhotoEditingContextMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An editing session for modifying the photo, video, and audio content of a Live Photo.
Notes: A Live Photo is a picture, captured by a supported iOS device, that includes motion and sound from the moments just before and after it was taken. Editing the content of a Live Photo works much like editing other asset types:

1. In an app using the Photos framework, fetch a PHAsset object that represents the Live Photo to edit, and use that object,Äôs requestContentEditingInputWithOptions method to retrieve a PHContentEditingInputMBS object.
In a photo editing extension that runs within the Photos app, your extension,Äôs main view controller (which adopts the PHContentEditingController protocol) receives a PHContentEditingInputMBS object when the user chooses to edit a Live Photo with your extension.
2. Create a Live Photo editing context with the initWithLivePhotoEditingInput initializer.
You can create a Live Photo editing context only from PHContentEditingInputMBS object that represents a Live Photo. Use the livePhoto property of the editing input to verify that it has live Photo content.
3. Use the frameProcessor property to define a block to be used in processing the Live Photo,Äôs visual content. Photos will call this block repeatedly to process each frame of the Live Photo,Äôs video and still photo content.
4. Create a PHContentEditingOutputMBS object to store the results of your edit, then call the saveLivePhotoToOutput:options to process the Live Photo and save it to your editing output object. This method applies your frameProcessor to each frame.
Note
You can also use the prepareLivePhotoForPlaybackWithTargetSize method to process a preview-quality version of the Live Photo to display in your app,Äôs UI during editing.
5. To allow a user to continue working with the edit later (for example, to adjust the parameters of a filter), create a PHAdjustmentDataMBS object describing your changes, and store it in the adjustmentData property of your editing output.
6. In an app using the Photos framework, use a photo library change block to commit the edit. (For details, see PHPhotoLibrary.) In the block, create a PHAssetChangeRequestMBS object and set its contentEditingOutput property to the editing output that you created.
In a photo editing extension, provide the PHContentEditingOutputMBS object that you created in your main view controller,Äôs finishContentEditing method.

When you use either of the methods listed in Processing an Editing Context,Äôs Live Photo, Photos calls your frameProcessor delegate repeatedly to process each frame of the Live Photo,Äôs video and still photo content. In that block, a PHLivePhotoFrameMBS object provides the Live Photo,Äôs existing content as a CIImageMBS object. You use Core Image to modify the image, then provide the result of your edits by returning a CIImageMBS object representing the result of processing the input image.

Core Image provides several ways to process the Live Photo,Äôs visual content. You can use the built-in filters listed in Core Image Filter Reference or create CIFilterMBS subclasses using custom graphics kernel

code. Or, to use other image processing technologies, you can directly access and modify image content in pixel buffers, Metal textures, or IOSurfaceRef objects with a custom CIImageProcessorKernel subclass.

12.28.2 Methods

12.28.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.12 or newer.

12.28.4 cancel

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Aborts any Live Photo processing in progress.
Notes: This method applies only after you,Äôve begun processing a Live Photo for display or output with the initWithLivePhotoEditingInput or saveLivePhotoToOutput method. After you call this method, Photos calls your completion handler and provides an error with the PHLivePhotoEditingErrorCodeAborted error code.

12.28.5 Constructor(livePhotoInput as PHContentEditingInputMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a Live Photo editing context for the specified editing input.
Notes: livePhotoInput: A content editing input object representing the Live Photo for which to perform editing.

Returns a new Live Photo editing context, or nil if the provided content editing input does not represent a Live Photo.

In an app using the Photos framework, you obtain a PHContentEditingInputMBS object by calling requestContentEditingInputWithOptions method of a PHAssetMBS object that you,Äôve previously fetched.
In a photo editing extension that runs within the Photos app, your extension,Äôs main view controller (which adopts the PHContentEditingController protocol) receives a PHContentEditingInputMBS object when the user chooses to edit a Live Photo with your extension.

You can create a Live Photo editing context only from PHContentEditingInputMBS object that represents a Live Photo. Use the livePhoto property of the editing input to verify that it has live Photo content.

12.28.6 PHLivePhotoShouldRenderAtPlaybackTime as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether processing should occur during or before playback.
Notes: The value for this key is an NSNumber object with a Boolean value. With the default value of false, Photos always renders your edits immediately when you call the Constructor, calling your frameProcessor block for each frame in the Live Photo,Äôs video and still photo content.

When you specify a value of true, Photos can defer rendering until playback time, calling your frameProcessor block only for photo and video frames that need to be displayed. However, in this case Photos may still choose to pre-render your edits if needed.

This option does not apply when rendering for output with the saveLivePhotoToOutput method.

12.28.7 prepareLivePhotoForPlayback(targetSize as CGSizeMBS, options as Dictionary = nil, completionHandler as PrepareLivePhotoForPlaybackCompletedMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Processes a Live Photo with your edits for viewing.
Notes: targetSize: The size of the view in which you plan to preview the edited Live Photo output.
options: Options that affect Live Photo rendering. See Live Photo Processing Options.

Use this method to generate preview versions of the edited Live Photo—for example, to display in your editing UI.

12.28.8 saveLivePhotoToOutput(ContentEditingOutput as PHContentEditingOutputMBS, options as Dictionary = nil, completionHandler as PrepareLivePhotoForPlaybackCompletedMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Processes and saves a full-quality Live Photo as the output of your editing session.
Notes: output: The photo editing output to receive the rendered Live Photo, created from the same PHContentEditingInputMBS object you used to begin this Live Photo editing context.
options: Options that affect Live Photo rendering. See Live Photo Processing Options.

Use this method when you have finished an editing session and need to provide rendered output in a PHContentEditingOutputMBS object. Unlike when rendering output for a photo or video asset, you don,Äôt need to provide rendered output using the renderedContentURL property of the editing output object. Instead, create a PHContentEditingOutputMBS object using the Constructor initializer, passing the same PHContentEditingInputMBS object you used in the FrameProcessing delegate initializer to start this Live Photo editing context. Then pass that editing output object to this method, and Photos renders the Live Photo and provides it to the editing output.

Don,Äôt forget to describe your edits in a PHAdjustmentDataMBS object and provide that to the adjustmentData property of your content editing output. Providing adjustment data allows your app (or photo editing extension) to non-destructively resume working with an edit later, whether on the same device or on another Mac or iOS device using iCloud Photo Library.

After this method,Äôs completion handler signals successful rendering, you use the content editing output to complete the edit. In an app using the Photos framework, create a PHAssetChangeRequestMBS object inside a changes, and set its contentEditingOutput property to your editing output. In a photo editing extension running in the Photos app, your main view controller provides content editing output when requested by the finishContentEditing method.

12.28.9 SetFrameProcessor(FrameProcessor as FrameProcessorMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets a delegate to be called by Photos for processing each frame of the Live Photo,Äôs visual content.
Notes: Use this property to define the image processing to be performed on each frame of the Live Photo. Setting this property does not begin processing; instead, after you call one of the methods listed in Processing an Editing Context,Äôs Live Photo, Photos executes your block repeatedly to process each frame of the Live Photo,Äôs video and still photo content.

In your frame processor block, use the image property of the provided PHLivePhotoFrameMBS object to access the image to be processed, and return a CIImageMBS object representing the result of your processing. For example, the following code sets up a processor block to apply a simple sepia-tone filter, then calls the saveLivePhotoToOutput method to begin processing the Live Photo for output.

12.28.10 Properties

12.28.11 audioVolume as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The audio gain to apply to the processed Live Photo.
Notes: Values must be between 0.0 and 1.0, inclusive. A value of 1.0 (the default) leaves the audio content of the Live Photo unchanged. A value of 0.0 mutes all audio in the output Live Photo.
Setting this property does not process the Live Photo content; instead, it sets the audio gain to be applied when you later process the Live Photo using one of the methods listed in Processing an Editing Context,Äôs Live Photo.
(Read and Write property)

12.28.12 duration as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The duration, in seconds, of the Live Photo.
Notes: (Read only property)

12.28.13 fullSizeImage as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The unedited still photo content of the Live Photo.
Notes: Value is CIImageMBS.
Use this property to display the unedited Live Photo,Äôs photo content, or to provide still-image previews in your editing UI.

For the best interactive editing performance, use a Metal, OpenGL, or OpenGL ES view and corresponding Core Image context to render editing results. See Processing Images in Core Image Programming Guide for details and examples.

This image does not reflect the Live Photo,Äôs orientation metadata. Adjust your rendering based on the orientation property to ensure that the image appears to the user in the correct orientation.
(Read only property)

12.28.14 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.28.15 orientation as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The image orientation of the Live Photo.
Notes: This fullSizeImage object does not reflect the Live Photo,Äôs orientation metadata. Use this property when displaying that image to ensure that it appears to the user in the correct orientation.
(Read only property)

12.28.16 photoTime as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The offset, in seconds, from the beginning of the Live Photo,Äôs duration to the time corresponding to its still photo.
Notes: To create time-based effects in your frameProcessor block, use this property together with the duration property.
(Read only property)

12.28.17 Delegates

12.28.18 FrameProcessorMBS(frame as PHLivePhotoFrameMBS, byref error as NSErrorMBS, tag as Variant) as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate for a block Photos calls to process Live Photo frames.
Notes: Result of delegate method must be a CIImageMBS object.

To apply edits to a Live Photo, define a block with this signature and assign it to the frameProcessor property of a Live Photo editing context. Then call the initWithLivePhotoEditingInput: to prepare a preview-quality version of your edits for display, or the saveLivePhotoToOutput method to produce full-quality final output. When you call one of those methods, Photos calls your frame processor block repeatedly—processing each frame of the Live Photo,Äôs video content as well as its still photo content—to render the output.

This delegates takes the following parameters:

frame: A PHLivePhotoFrameMBS object describing the frame image to be processed.
error: If your delegate cannot successfully process the frame, set this to an error object describing the failure.

Your block should return a CIImageMBS object representing the result of your edits, or nil to indicate that your image processing has failed and the Live Photo edit should be aborted. Use the frame parameter,Äôs

image property to access the image to be edited.

12.28.19 PrepareLivePhotoForPlaybackCompletedMBS(options as Dictionary, livePhoto as PHLivePhotoMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls on the main queue after processing is complete.
Notes: livePhoto: The rendered Live Photo, suitable for displaying in a PHLivePhotoViewMBS object.
error: If preparing the edited Live Photo for display succeeds, this parameter is nil. If processing fails, the livePhoto parameter is nil, and this parameter contains an error object describing the failure.

12.28.20 SaveLivePhotoToOutputCompletedMBS(output as PHContentEditingOutputMBS, options as Dictionary, success as Boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls on the main queue after rendering is complete.
Notes: success: True if rendering succeeds; otherwise false.
error: If rendering succeeds, this parameter is nil. If rendering fails, this parameter contains an error object describing the failure.

12.29 class PHLivePhotoFrameMBS

12.29.1 class PHLivePhotoFrameMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A container that provides image content for a single frame of a Live Photo in an editing context.
Notes: You don,Äôt create classes that implement this protocol. Instead, you provide a frameProcessor block when editing a Live Photo with the PHLivePhotoEditingContextMBS class. When you process your edits for output or display, Photos calls your block repeatedly to process each frame of the Live Photo,Äôs video and still photo content. On each call, Photos provides the frame,Äôs image content and associated information in an object that adopts this protocol. In that block, you use that object,Äôs image property to access the image to be edited, then perform your edits and return another CIImageMBS object representing the result of processing the input image.

Requires macOS 10.12 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.29.2 Methods

12.29.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.12 or newer.

12.29.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.29.5 Properties

12.29.6 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.29.7 Image as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The image content of the frame to be processed.
Notes: Value is CIImageMBS object.
Core Image provides several ways to perform your adjustments to this image:

- Create a single CIFilterMBS object, or a chain of filters. Set this image as the kCIInputImageKey parameter of the first filter in the chain, and use the last filter,Äôs outputImage property to access the result. Each CIFilter object can be a built-in filter or a custom filter subclass that you create.
- Use the imageByApplyingFilter method to conveniently apply one of the many built-in Core Image filters.
- Access pixel buffers directly and apply custom image processing using a custom CIImageProcessorKernelMBS subclass.

In all cases, you obtain another CIImageMBS object representing the result of your adjustments. Return that image from your frameProcessor block. (See the frameProcessor description for example code.)
(Read only property)

12.29.8 renderScale as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The scale factor of the frame image relative to the Live Photo,Äôs photo content.
Notes: Photos calls your frameProcessor block repeatedly, both to process each frame of the Live Photo,Äôs video content and to process the Live Photo,Äôs still photo content. Video frames can be a different size than still photo content—use this value to scale any of your image processing parameters that depend on the image,Äôs size.
(Read only property)

12.29.9 time as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The time offset, in seconds, of this frame relative to the start of the Live Photo.
Notes: You can use this value to vary your image processing over time, creating animated effects.
(Read only property)

12.29.10 Type as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The type of image content in this frame.
Notes: Photos calls your frameProcessor block repeatedly, both to process each frame of the Live Photo,Äôs video content and to process the Live Photo,Äôs still photo content. Use this property to distinguish photo content from video frames—for example, to add a watermark only to still photo content.
(Read only property)

12.29.11 Constants

Types

Constant	Value	Description
TypePhoto	0	The image is a still photo.
TypeVideo	1	The image is a single frame from the Live Photo,Äôs video content.

12.30 class PHLivePhotoMBS

12.30.1 class PHLivePhotoMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A displayable representation of a Live Photo—a picture that includes motion and sound from the moments just before and after its capture.
Notes: In macOS, you can use this class to display edits in progress for Live Photo content in a photo editing extension.

For guidance on how to integrate Live Photos with your app,Äôs user experience, see Live Photos in iOS Human Interface Guidelines.

The PHLivePhotoMBS class serves in much the same role for Live Photos as the NSImageMBS class serves for static images. A NSImageMBS object represents not the data file an image is loaded from, but instead a ready-to-use image that can be displayed in a view—similarly, a PHLivePhotoMBS object represents a Live Photo ready to display with motion and sound using a PHLivePhotoViewMBS object, not an entry in the Photos library or the data resources that constitute a Live Photo. (To work with Live Photos as elements of the Photos library, use the PHAsset class. To work with the data files that constitute a Live Photo, use the PHAssetResourceMBS class.)
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- The Top 10 from the MBS Xojo Plugins in 2022
- News from the MBS Xojo Plugins Version 22.5

12.30.2 Methods

12.30.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.12 or newer.

12.30.4 cancelLivePhotoRequestWithRequestID(requestID as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels an asynchronous request.
Notes: requestID: The numeric identifier of the request to be canceled.

When you use the requestLivePhotoWithResourceFileURLs method to asynchronously load a Live Photo from resource files, the method returns a numeric identifier for the request. To cancel the request before it completes, provide the identifier when calling the cancelLivePhotoRequestWithRequestID method.

12.30.5 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.30.6 copy as PHLivePhotoMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

12.30.7 PHLivePhotoInfoCancelledKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary in RequestLivePhotoWithResourceFileURLsMBS delegate.
Notes: A Boolean value indicating whether the Live Photo loading request was canceled.

12.30.8 PHLivePhotoInfoErrorKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary in RequestLivePhotoWithResourceFileURLsMBS delegate.
Notes: An error that occurred while attempting to load the requested Live Photo.

12.30.9 PHLivePhotoInfoIsDegradedKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys for the info dictionary in RequestLivePhotoWithResourceFileURLsMBS delegate.

Notes: A Boolean value indicating whether the result Live Photo is a low-quality substitute for the requested Live Photo.

If true, the result parameter of your resultHandler block contains a still or low-quality Live Photo, and Photos will call your result delegate method again to provide the full motion and sound content of the Live Photo. If false, Photos has provided all possible data and will not call your result handler again.

12.30.10 requestLivePhotoWithResourceFileURLs(Files() as FolderItem, placeholderImage as NSImageMBS = nil, targetSize as CGSizeMBS, contentMode as Integer, theDelegate as RequestLivePhotoWithResourceFileURLsMBS, Tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Asynchronously loads a Live Photo from the specified resource files.
Notes: fileURLs: An array of Folderitems containing the resource URLs that constitute a Live Photo, as obtained using the PHAssetResource class.
image: A static image to represent the Live Photo before its full content has been loaded and validated.
targetSize: The target size of Live Photo to be returned. Pass zero size to obtain the requested Live Photo at its original size.
contentMode: An option for how to fit the image to the aspect ratio of the requested size. For details, see PHLivePhotoMBS class.
See also:

- 12.30.11 requestLivePhotoWithResourceFileURLs(fileURLs() as String, placeholderImage as NSImageMBS = nil, targetSize as CGSizeMBS, contentMode as Integer, theDelegate as RequestLivePhotoWithResourceFileURLsMBS, Tag as Variant = nil) as Integer 828

12.30.11 requestLivePhotoWithResourceFileURLs(fileURLs() as String, placeholderImage as NSImageMBS = nil, targetSize as CGSizeMBS, contentMode as Integer, theDelegate as RequestLivePhotoWithResourceFileURLsMBS, Tag as Variant = nil) as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Asynchronously loads a Live Photo from the specified resource files.
Notes: fileURLs: An array of Folderitems containing the resource URLs that constitute a Live Photo, as obtained using the PHAssetResource class.
image: A static image to represent the Live Photo before its full content has been loaded and validated.
targetSize: The target size of Live Photo to be returned. Pass zero size to obtain the requested Live Photo at its original size.
contentMode: An option for how to fit the image to the aspect ratio of the requested size. For details, see PHLivePhotoMBS class.
See also:

12.30.12 Properties

12.30.13 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.30.14 Height as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The size, in pixels, of the Live Photo.
Notes: (Read only property)

12.30.15 Width as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The size, in pixels, of the Live Photo.
Notes: (Read only property)

12.30.16 Constants

Constants

Constant	Value	Description
RequestIDInvalid	0	The value for an invalid request ID.

Content Modes

Constant	Value	Description
ContentModeAspectFill	1	Scales the image so that it completely fills the target size.
ContentModeAspectFit	0	Scales the image so that its larger dimension fits the target size.
ContentModeDefault	0	Fits the image to the requested size using the default option, ContentModeAspectFit.

12.30.17 Delegates

12.30.18 RequestLivePhotoWithResourceFileURLsMBS(LivePhoto as PHLivePhotoMBS, placeholderImage as NSImageMBS, targetSize as CGSizeMBS, contentMode as Integer, tag as variant, info as Dictionary)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The method called when live photo is loaded.
Notes: LivePhoto: The requested Live Photo object.
info: A dictionary providing information about the status of the request. See PHLivePhotoMBS class for possible keys and values.

Method may be called twice if there is a low res and high res version.

12.31 class PHLivePhotoRequestOptionsMBS

12.31.1 class PHLivePhotoRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the delivery of Live Photo assets you request from an image manager.
Notes: A Live Photo is a picture that includes movement and sound from the moments just before and after its capture.
Available in MacOS 10.15 or newer.

12.31.2 Methods

12.31.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.31.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.31.5 copy as PHLivePhotoRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the options object.

12.31.6 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.31.7 SetProgressHandler(ProgressHandler as ProgressHandlerMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the delegate which Photos calls periodically while downloading the Live Photo.
Notes: If you request a Live Photo whose data is not on the local device, and you have enabled downloading with the PHLivePhotoRequestOptionsMBS property, Photos calls your block periodically to report progress and to allow you to cancel the download.

12.31.8 Properties

12.31.9 DeliveryMode as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The requested Live Photo quality and delivery priority.
Notes: Use this property to tell Photos to provide a Live Photo quickly (possibly sacrificing image quality), to provide a high-quality Live Photo (possibly sacrificing speed), or to provide both automatically if needed. See DeliveryMode constants.
(Read and Write property)

12.31.10 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.31.11 NetworkAccessAllowed as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether Photos can download the requested Live Photo data from iCloud.
Notes: If true, and the requested Live Photo data is not stored on the local device, Photos downloads that data from iCloud. To be notified of the download,Äôs progress, use the progressHandler property to provide a block that Photos calls periodically while downloading. If false (the default), and the Live Photo data is not on the local device, the PHImageResultIsInCloudKey value in the result handler,Äôs info dictionary indicates that the data is not available unless you enable network access.
(Read and Write property)

12.31.12 ProgressHandler as ProgressHandlerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate which Photos calls periodically while downloading the Live Photo.
Notes: If you request a Live Photo whose data is not on the local device, and you have enabled downloading with the PHLivePhotoRequestOptionsMBS property, Photos calls your block periodically to report progress and to allow you to cancel the download.
(Read and Write property)

12.31.13 Version as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The version of the Live Photo to be requested.
Notes: Use this property to request a version of the Live Photo with or without adjustments. See Version constants.
(Read and Write property)

12.31.14 Constants

Delivery Modes

Constant	Value	Description
DeliveryModeFastFormat	2	Photos provides only a fast-loading image, possibly sacrificing image quality.
DeliveryModeHighQualityFormat	1	Photos provides only the highest-quality image available, regardless of hov much time it takes to load.
DeliveryModeOpportunistic	0	Photos automatically provides one or more results in order to balance imag quality and responsiveness.

Version Type

Constant	Value	Description
VersionCurrent	0	Request the most recent version of the image asset (the one that reflects all edits).
VersionOriginal	2	Request the original, highest-fidelity version of the image asset.
VersionUnadjusted	1	Request a version of the image asset without adjustments.

12.31.15 Delegates

12.31.16 ProgressHandlerMBS(progress as double, error as NSErrorMBS, byref stop as boolean, info as Dictionary, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls periodically while downloading the Live Photo..
Notes: If you request a Live Photo whose data is not on the local device, and you have enabled downloading with the progressHandler property in PHLivePhotoRequestOptionsMBS class, Photos calls your block periodically to report progress and to allow you to cancel the download.

progress: A floating-point value indicating the progress of the download. A value of 0.0 indicates that the download has just started, and a value of 1.0 indicates the download is complete.
error: An NSErrorMBS object describing an error that occurred when attempting to download the image, or nil if no errors have occurred.
stop: A Boolean value. To cancel the download, set stop to true inside the method.
info: A dictionary providing additional information about the status of the image request. See Image Result Info Keys for possible keys and values. For example, a true value for the key PHImageResultIsDegradedKey indicates that Photos may send the full-quality version of the image later, depending on the value of the image request,Äôs deliveryMode property.

If this method is dispatched asynchronously to the main thread, the stop property doesn't work.

12.32 class PHObjectChangeDetailsMBS

12.32.1 class PHObjectChangeDetailsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A description of changes that occurred in an asset or collection object.
Notes: A PHObjectChangeDetailsMBS object provides detailed information about differences between two states of an asset or collection object—one that you previously obtained and an updated state that would result if you fetched that entity again. You observe changes via events in PHPhotoLibraryMBS object. When Photos notifies your observer of a change, you get change details by passing the object you,Äôre interested in to the changeDetailsForObject method.

For an asset collection or collection list, a PHObjectChangeDetailsMBS object describe changes only to the collection,Äôs properties. If you are instead interested in changes to the collection,Äôs membership, fetch the collection,Äôs contents and use the changeDetailsForFetchResult: method to track changes to the fetch result.

Needs macOS 10.13 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.32.2 Methods

12.32.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.32.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.32.5 Properties

12.32.6 assetContentChanged as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the asset,Äôs photo or video content has changed.
Notes: If this value true, you can use the PHImageManagerMBS class to retrieve updated content.
This value is false if the asset has not changed or if the change details do not refer to a PHAssetMBS object.
(Read only property)

12.32.7 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.32.8 objectAfterChanges as PHObjectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that reflects the current state of the asset or collection it represents.
Notes: Read this object,Äôs properties to retrieve updated metadata for the asset or collection. Using this object is equivalent to repeating the same fetch that returned the original object.
(Read only property)

12.32.9 objectBeforeChanges as PHObjectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that reflects the original state of the asset or collection it represents.
Notes: This property,Äôs value is the same object you passed to the changeDetailsForObject to request change details.
(Read only property)

12.32.10 objectWasDeleted as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the object has been deleted from the Photos library.

Notes: If this value is true, the asset or collection has been permanently deleted from the Photos library.

To instead track the removal of assets from collections (or collections from collection lists), fetch the collection,Äôs contents and use the changeDetailsForFetchResult method to track changes to the fetch result. (Read only property)

12.33 class PHObjectMBS

12.33.1 class PHObjectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The abstract superclass for Photos model objects (assets and collections).
Notes: You do not create or use instances of this class directly. Instead, work with instances of its concrete subclasses —PHAssetMBS, PHAssetCollectionMBS, PHCollectionListMBS, and PHObjectPlaceholderMBS.

Because the PHObjectMBS class implements the isEqual and hash methods in terms of its localIdentifier property, you can use techniques that depend on these methods to keep track of asset and collection objects.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.33.2 Methods

12.33.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.33.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.33.5 copy as PHObjectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

12.33.6 Properties

12.33.7 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.33.8 localIdentifier as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A unique string that persistently identifies the object.
Notes: Use this string to find the object by using the fetchAssetsWithLocalIdentifiers, fetchAssetCollectionsWithLocalIdentifiers, or fetchCollectionListsWithLocalIdentifiers method.
(Read only property)

12.34 class PHObjectPlaceholderMBS

12.34.1 class PHObjectPlaceholderMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A read-only proxy representing a Photos asset or collection object yet to be created by a change request.
Notes: You obtain object placeholders when you use change requests to create assets, collections, or collection lists. After the change request completes, you can use the object placeholder to fetch the newly created object. You can also use an object placeholder to make additional change requests involving the object to be created.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

12.34.2 Methods

12.34.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.34.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

12.34.5 Properties

12.34.6 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.35 class PHPhotoLibraryMBS

12.35.1 class PHPhotoLibraryMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A shared object that manages access and changes to the user‚Äôs shared photo library.
Notes: The shared PHPhotoLibraryMBS object represents the entire set of assets and collections managed by the Photos app, including both assets stored on the local device and (if enabled) those stored in iCloud Photos. You use this object for the following tasks:

- Getting or verifying the user's permission for your app to access Photos content.
- Making changes to assets and collections; for example, editing asset metadata or content, inserting new assets, or rearranging the members of a collection.
- Registering for update messages sent when changes are made to the library.

Requires MacOS 10.13 or newer.
Blog Entries

- New in the MBS Xojo Plugins Version 20.2
- Show Live Photos in your Xojo application

Videos

- XDC 2020 MBS Plugins Presentation

12.35.2 Methods

12.35.3 authorizationStatus as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns information about your app‚Äôs authorization to access the user‚Äôs photo library.
Notes: Returns the current authorization status.

Accessing the photo library always requires explicit permission from the user. The first time your app uses PHAssetMBS, PHCollectionMBS, PHAssetCollectionMBS, or PHCollectionListMBS methods to fetch content from the library, or uses one of the methods listed in Applying Changes to the Photo Library to request changes to library content, Photos automatically and asynchronously prompts the user to request authorization.

Important
Your app,Äôs Info.plist file must provide a value for the NSPhotoLibraryUsageDescription key that explains to the user why your app is requesting Photos access. Apps linked on or after iOS 10 will crash if this key is not present.

After the user grants permission, the system remembers the choice for future use in your app, but the user can change this choice at any time using the Settings app. If the user has denied your app photo library access, not yet responded to the permission prompt, or cannot grant access due to restrictions, any attempts to fetch photo library content will return empty PHFetchResultMBS objects, and any attempts to perform changes to the photo library will fail.

Tip
Use the DidChange event to observe photo library changes before fetching content. After the user grants access to the photo library for your app, Photos sends change messages for any empty fetch results you retrieved beforehand, notifying you that library content for those fetches is now available.

If this method returns PHAuthorizationStatusNotDetermined, you can call the requestAuthorization method to prompt the user for photo library access permission.

12.35.4 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.35.5 cloudIdentifiersForLocalIdentifiers(localIdentifiers() as String) as PHCloudIdentifierMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Queries cloud identifiers for local identifiers.

12.35.6 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.35.7 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.35.8 localIdentifiersForCloudIdentifiers(cloudIdentifiers() as PHCloudIdentifierMBS) as String()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Queries local identifiers for cloud identifiers.

12.35.9 performChanges(ChangeBlock as ChangeBlockMBS, CompletionHandler as ChangeCompletionHandlerMBS, tag as Variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Asynchronously runs a block that requests changes to be performed in the photo library.
Notes: We call back to main thread for delegates, so please keep main thread available.

12.35.10 PHLocalIdentifierNotFound as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the error constants.
Notes: The local identifier was not found.

12.35.11 PHPhotosErrorDomain as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The error domain for Photos errors.

12.35.12 requestAuthorization(CompletionHandler as RequestAuthorizationCompletedMBS, tag as Variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Requests the user,Äôs permission, if needed, to access the photo library.
Notes: CompletionHandler: Photos calls this delegate upon determining your app,Äôs authorization to access the photo library.

Accessing the photo library always requires explicit permission from the user. The first time your app uses PHAssetMBS, PHCollectionMBS, PHAssetCollectionMBS, or PHCollectionListMBS methods to fetch content from the library, or uses one of the methods listed in Applying Changes to the Photo Library to request changes to library content, Photos automatically and asynchronously prompts the user to request authorization. Alternatively, you can call this method to prompt the user at a time of your choosing.

Important:
Your app,Äôs Info.plist file must provide a value for the NSPhotoLibraryUsageDescription key that explains to the user why your app is requesting Photos access. Apps linked on or after iOS 10 will crash if this key is not present.

After the user grants permission, the system remembers the choice for future use in your app, but the user can change this choice at any time using the Settings app. If the user has denied your app photo library access, not yet responded to the permission prompt, or cannot grant access due to restrictions, any attempts to fetch photo library content will return empty PHFetchResultMBS objects, and any attempts to perform changes to the photo library will fail.

Use the DidChange event to observe photo library changes before fetching content. After the user grants access to the photo library for your app, Photos sends change messages for any empty fetch results you retrieved beforehand, notifying you that library content for those fetches is now available.

This method always returns immediately. If the user has previously granted or denied photo library access permission, it executes the delegate when called; otherwise, it displays an alert and executes the block only after the user has responded to the alert.

12.35.13 sharedPhotoLibrary as PHPhotoLibraryMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Retrieves the shared photo library object.
Notes: The singleton photo library object.

12.35.14 Properties

12.35.15 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.35.16 unavailabilityReason as NSErrorMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The error object to give reason for unavailability.
Notes: (Read only property)

12.35.17 Events

12.35.18 DidBecomeUnavailable

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: The Photos library did become unavailable.

12.35.19 DidChange(changes as PHChangeMBS)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function:

Tells you that a set of changes has occurred in the Photos library.
Required.
Notes:

change: An object representing the changes.

Use the provided PHChangeMBS object to find out which, if any, of the albums or collections you,Äôre interested in have changed and get detailed change information. Call the change object,Äôs changeDetailsForObject method to get information about changes to an asset,Äôs contents or metadata properties or about a collection,Äôs metadata properties. Call the change object,Äôs changeDetailsForFetchResult to get information about changes to a collection,Äôs list of members (or to any other fetch result).

12.35.20 Constants

Authorization Status

Constant	Value	Description
PHAuthorizationStatusAuthorized	3	The user has explicitly granted your app access to the photo library.
PHAuthorizationStatusDenied	2	The user has explicitly denied your app access to the photo library.
PHAuthorizationStatusNotDetermined	0	Explicit user permission is required for photo library access, but the us not yet granted or denied such permission.
PHAuthorizationStatusRestricted	1	Your app is not authorized to access the photo library, and the user c grant such permission.

Errors

Constant	Value	Description
PHPhotosErrorInvalid	-1	Invalid error?
PHPhotosErrorLibraryVolumeOffline	3114	Library volume offline.
PHPhotosErrorRelinquishingLibraryBundleToWriter	3142	Relinquishing library bundle to writer
PHPhotosErrorSwitchingSystemPhotoLibrary	3143	Failed to switch system photo library.
PHPhotosErrorUserCancelled	3072	The user cancelled.

12.35.21 Delegates

12.35.22 ChangeBlockMBS(tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that requests changes to be performed.
Notes: This delegate takes no parameters and has no return value.

12.35.23 ChangeCompletionHandlerMBS(success as boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A delegate that Photos calls after the change block completes and Photos performs the requested changes.
Notes: success: True if Photos successfully applied the changes requested in the block; otherwise, false.
error: If an error occurs, an NSError object describing the error; otherwise, nil.

12.36 class PHProjectChangeRequestMBS

12.36.1 class PHProjectChangeRequestMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A request to change asset data in a Photos project extension.
Notes: Make a project change request to alter a project's title or metadata. Respond to project change requests by updating your user interface as assets are added, modified, or removed.

Available in macOS 10.13 or newer.
Subclass of the PHChangeRequestMBS class.

12.36.2 Methods

12.36.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.36.4 Constructor(project as PHProjectMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Creates a change request around the specified project.
Notes: project: The project being changed in the request.

12.36.5 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

12.36.6 removeAssets(assets() as PHAssetMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Removes the specified assets from the project.
Notes: assets: A collection of PHAssets to be removed from the project.

12.36.7 Properties

12.36.8 keyAsset as PHAssetMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Sets the key asset representing the project.
Notes: Setting a key asset has been deprecated in macOS 10.14. Use ProjectPreviewImage to provide a rendered preview image instead of designating a key asset.
(Read and Write property)

12.36.9 projectExtensionData as Memoryblock

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Compressed project-specific data to use in the change request.
Notes: The total size of stored data is limited to 1 MB; attempting to store more data will result in an error. Don't include rasterized images that can be locally cached. Limit stored data to compressed project-specific data.
(Read and Write property)

12.36.10 projectPreviewImage as NSImageMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The project preview in Photos.
Notes: When setting, please pass a rendered project preview with dimensions of 1024 x 1024.

PhotoKit requires that you set a project preview in the following situations:

- Whenever the project changes in a way that requires a new preview.
- During or after execution of the beginProjectWithExtensionContext protocol method.
- During or after execution of the resumeProjectWithExtensionContext protocol method.

(Read and Write property)

12.36.11 title as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The title of the change request.
Notes: (Read and Write property)

12.37 class PHProjectMBS

12.37.1 class PHProjectMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A representation of a Photos app project extension.
Notes: This class represents the project when extended from macOS Photos. Projects can have the following types:

- Book
- Calendar
- Card
- Prints
- Slideshow
- Wall decor

Users create projects by selecting one or more assets, right-clicking the selection, and grouping the assets, much like an album collection. Your app treats the project as a separate entity, represented as a PHProject. Subclass of the PHAssetCollectionMBS class.

12.37.2 Methods

12.37.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.13 or newer.

12.37.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: The constructor.

12.37.5 Properties

12.37.6 hasProjectPreview as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: A property that indicates whether a project preview was previously set.
Notes: To set the preview, use ProjectPreviewImage property.
(Read only property)

12.37.7 projectExtensionData as Memoryblock

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Data associated with the project extension.
Notes: (Read only property)

12.38 class PHVideoRequestOptionsMBS

12.38.1 class PHVideoRequestOptionsMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A set of options affecting the delivery of video asset data that you request from an image manager.
Notes: Available on MacOS 10.15 or newer.

12.38.2 Methods

12.38.3 available as boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether class is available.
Notes: Should return true on MacOS 10.15 or newer.

12.38.4 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

12.38.5 Destructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

12.38.6 SetProgressHandler(ProgressHandler as ProgressHandlerMBS, tag as variant = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the delegate that Photos calls periodically while downloading the video.
Notes: If you request a video whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your delegate periodically to report progress and to allow you to cancel the download.

12.38.7 Properties

12.38.8 DeliveryMode as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mode specifying the requested video quality and delivery priority.
Notes: Use this property to tell Photos to provide a video quickly (possibly sacrificing image quality) or to provide a high-quality video (possibly sacrificing speed). This option applies only when requesting the current version of the video (that is, only when the version property is VersionCurrent).
The default option is DeliveryModeAutomatic. See DeliveryMode.
(Read and Write property)

12.38.9 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

12.38.10 NetworkAccessAllowed as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether Photos can download the requested video from iCloud.
Notes: If true, and the requested video is not stored on the local device, Photos downloads the video from iCloud. To be notified of the download,Äôs progress, use the progressHandler property to provide a delegate that Photos calls periodically while downloading the video. If false (the default), and the video is not on the local device, the PHImageResultIsInCloudKey value in the result handler,Äôs info dictionary indicates that the video is not available unless you enable network access.
(Read and Write property)

12.38.11 ProgressHandler as ProgressHandlerMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls periodically while downloading the video.
Notes: If you request a video whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your delegate periodically to report progress and to allow you to cancel the download.
(Read and Write property)

12.38.12 Version as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The version of the video to request.
Notes: If a video asset has been edited, use this property to request a video with or without adjustments. The default option is VersionCurrent. See Version constants.
(Read and Write property)

12.38.13 Constants

Delivery Modes

Constant	Value	Description
DeliveryModeAutomatic	0	Photos automatically determines which quality of video data to provide b on the request and current conditions.
DeliveryModeFastFormat	3	Photos provides whatever quality of video can be most quickly loaded.
DeliveryModeHighQualityFormat	1	Photos provides only the highest quality video available.
DeliveryModeMediumQualityFormat	2	Photos provides a video of moderate quality unless a higher quality versio locally cached.

Version Mode

Constant	Value	Description
VersionCurrent	0	Request the most recent version of the video asset, reflecting all edits. The resulting video is the rendered output from all previously made adjustments.
VersionOriginal	1	Request a version of the video asset without adjustments. The resulting video is the originally captured or imported version of the asset, regardless of any edits that have been made.

12.38.14 Delegates

12.38.15 ProgressHandlerMBS(progress as double, tag as Variant, error as NSErrorMBS, byref stop as Boolean, info as Dictionary)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate that Photos calls while downloading asset data from iCloud.
Notes: If you request a video asset whose data is not on the local device, and you have enabled downloading with the networkAccessAllowed property, Photos calls your block periodically to report progress and allow canceling the download.

The method takes the following parameters:

progress: A floating-point value indicating the progress of the download. A value of 0.0 indicates the download has just started, and a value of 1.0 indicates the download is complete.
error: An NSErrorMBS object describing an error that occurred when attempting to download the video, or nil if no errors have occurred.
stop: A pointer to a Boolean value. To cancel the download, set stop to true inside the method.
info: A dictionary providing additional information about the status of the video request. See Image Result Info Keys for possible keys and values.

If the event is dispatched asynchronously to main thread, the stop property will not work.

Chapter 13

SceneKit

13.1 control DesktopSCNControlMBS

13.1.1 control DesktopSCNControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a SCNViewMBS.
Notes: This control embeds a special SCNView subclass.
Designed for Xojo 2021r3 and newer.
Please use view property to access the underlaying object and set properties.

To learn about rendering events, please check here:
https://developer.apple.com/documentation/scenekit/scnscenerendererdelegate?language=objc
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- Three new controls for iOS in Xojo
- MBS Xojo Plugins, version 20.6pr3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.2
- MBS Xojo Plugins, version 20.2pr1
- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 19.6, page 45: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes

- 18.4, page 39: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 26: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.1.2 Properties

13.1.3 View as SCNViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.
Notes: Use this object to set more options on the control.
(Read only property)

13.1.4 Events

13.1.5 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

13.1.6 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

13.1.7 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.
In older Xojo versions, this event is named LostFocus.
Notes:

This only fires if the control itself lost focus and not a sub control.

13.1.8 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.
In older Xojo versions, this event is named GotFocus.
Notes:

This only fires if the control itself got focus and not a sub control.

13.1.9 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

13.1.10 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named EnableMenuItems.

13.1.11 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown.
If you return False, the system handles the MouseDown so the above event handlers do not get called.

13.1.12 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

13.1.13 MouseUp(x As Integer, y As Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

13.1.14 rendererDidApplyAnimations(time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur after actions and animations are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this method to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.1.15 rendererDidApplyConstraints(time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked on the control once the scene renderer did apply the constraints.
Notes: time The time at which the constraints were simulated.

All modifications done within this method don't go through the transaction model, they are directly applied on the presentation tree.

13.1.16 rendererDidRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells the control that the renderer has rendered the scene.
Notes: scene: The scene object that was rendered.
time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this method to perform custom drawing after SceneKit has rendered a scene—for example, to draw overlay content on top of SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this method—the results of modifying SceneKit objects in this method are undefined.

- To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
- To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this method. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.1.17 rendererDidSimulatePhysics(time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur after physics simulations are performed.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this event exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

This event is the last opportunity SceneKit provides for you to change the scene graph before rendering.

13.1.18 rendererUpdate(time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur before actions, animations, and physics are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.1.19 rendererWillRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells the control that the renderer has cleared the viewport and is about to render the scene.
Notes: scene: The SCNSceneMBS object to be rendered.
time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this event to perform custom drawing before SceneKit renders a scene—for example, to draw backdrop content underneath SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this event—the results of modifying SceneKit objects during this event are undefined.

To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this event. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.1.20 ScaleFactorChanged(NewFactor as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

13.1.21 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

13.2 class SCNAccelerationConstraintMBS

13.2.1 class SCNAccelerationConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNAccelerationConstraint caps the acceleration and velocity of a node.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.2.2 Methods

13.2.3 accelerationConstraint as SCNAccelerationConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNAccelerationConstraint object.

13.2.4 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNAccelerationConstraint object.

13.2.5 Properties

13.2.6 damping as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the damping factor of the receiver.
Notes: Optionally reduce the body's linear velocity each frame to simulate fluid/air friction. Value should be zero or greater. Defaults to 0.1. Animatable.
(Read and Write property)

13.2.7 decelerationDistance as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Controls the distance at which the node should start decelerating.
Notes: Defaults to 0. Animatable.
(Read and Write property)

13.2.8 maximumLinearAcceleration as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Controls the maximum linear acceleration. Defaults to MAXFLOAT. Animatable.
Notes: The maximum linear acceleration is in m.s^-2
(Read and Write property)

13.2.9 maximumLinearVelocity as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Controls the maximum linear velocity. Defaults to MAXFLOAT. Animatable.
Notes: The maximum linear velocity is in m.s
(Read and Write property)

13.3 class SCNActionMBS

13.3.1 class SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A simple, reusable animation that changes attributes of any node you attach it to.
Notes: You use actions most often to change the structure and content of the SCNNodeMBS object to which they are attached, but you can also use actions make other changes to the scene. In SceneKit, actions provide an easy way to implement animated behaviors that frequently change in response to user input.

see also
https://developer.apple.com/documentation/scenekit/scnaction
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 19.6, page 50: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, page 51: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.4, pages 48 to 49: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 39 to 42: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.3.2 Methods

13.3.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.3.4 copy as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the action.

13.3.5 fadeIn(duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that changes the opacity of the node to 1.0.
Notes: duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs opacity property animates from its current value to 1.0.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.fadeOut(Duration)

13.3.6 fadeOpacityBy(factor as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that adjusts the opacity of a node by a relative value.
Notes: factor: The amount to change the node,Äôs opacity by.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs opacity property animates to its new value.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.fadeOpacityBy(-factor, duration)

13.3.7 fadeOpacityTo(opacity as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that adjusts the opacity of a node to a new value.
Notes: opacity: The new opacity value of the node.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs opacity property animates to its new value.
This action is not reversible; the reverse of this action has the same duration but does not change anything.

13.3.8 fadeOut(duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that changes the opacity of the node to 0.0.
Notes: duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs opacity property animates from its current value to 0.0.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.fadeIn(duration)

13.3.9 group(actions() as SCNActionMBS) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that runs a collection of actions in parallel.
Notes: actions: An array of SCNActionMBS objects.

Returns a new group action object.

When the action executes, the actions that make up the group all start immediately and run in parallel. The duration of the group action is the longest duration among the collection of actions. If an action in the group has a duration less than the group,Äôs duration, the action completes and then idles until the group completes the remaining actions. This matters most when creating a repeating action that repeats a group. This action is reversible; it creates a new group action that contains the reverse of each action specified in the group.

13.3.10 hide as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that hides a node.
Notes: When the action executes, the node,Äôs hidden property is set to true.

This action is reversible; the reverse is equivalent to the unhide action.

13.3.11 javaScriptActionWithScript(script as string, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that executes a JavaScript script periodically over a specified duration.
Notes: script: A string containing JavaScript source code.
seconds: The duration of the action, in seconds.

SceneKit exposes its classes, methods, and functions in the JavaScript context that runs the script—see the SCNJavaScript.h header file for details.

When the action executes, SceneKit runs the script repeatedly until the action,Äôs duration expires. Each time SceneKit runs the script, it computes the elapsed time since the action began executing (as a fraction of the action,Äôs duration between 0.0 and 1.0) and makes it available to the script as a variable named elapsedTime. The script can also reference the SCNNode object running the action as a variable named node.
This action is not reversible; the reverse action executes the same script.

13.3.12 moveBy(delta as SCNVector3MBS, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that moves a node relative to its current position.
Notes: delta: A vector that describes the change to be applied to the node,Äôs position.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs position property animates from its current position to its new position.
This action is reversible; the reverse is created as if the following code had been executed:

dim reverseDelta as SCNVector3MBS = SCNVector3MBS.Vector(-delta.x, -delta.y, -delta.z)
SCNActionMBS.moveBy(reverseDelta, duration)
See also:

- 13.3.13 moveBy(deltaX as double, deltaY as double, deltaZ as double, duration as double) as SCNAc-

13.3.13 moveBy(deltaX as double, deltaY as double, deltaZ as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that moves a node relative to its current position.
Notes: deltaX: The distance to move the node in the X direction of its parent node,Äôs local coordinate space.
deltaY: The distance to move the node in the Y direction of its parent node,Äôs local coordinate space.
deltaZ: The distance to move the node in the Z direction of its parent node,Äôs local coordinate space.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs position property animates from its current position to its new position.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.moveBy(-deltaX, -deltaY, -deltaZ, duration)
See also:

- 13.3.12 moveBy(delta as SCNVector3MBS, duration as double) as SCNActionMBS 869

13.3.14 moveTo(location as SCNVector3MBS, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that moves a node to a new position.
Notes: location: The coordinates for the node,Äôs new position in its parent node,Äôs local coordinate space.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs position property animates from its current position to its new position.
This action is not reversible; the reverse of this action has the same duration but does not move the node.

13.3.15 removeFromParentNode as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that removes the node from its parent.
Notes: When the action executes, the node is immediately removed from its parent.
This action is not reversible; the reverse of this action is the same action.

13.3.16 repeatAction(action as SCNActionMBS, count as integer) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that repeats another action a specified number of times.
Notes: action: The action to be executed.
count: The number of times to execute the action.

Returns a new action object.

When the action executes, the associated action runs to completion and then repeats, until the count is reached.
This action is reversible; it creates a new action that is the reverse of the specified action and then repeats it the same number of times.

13.3.17 repeatActionForever(action as SCNActionMBS) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that repeats another action forever.
Notes: action: The action to execute.

Returns a new action object.

When the action executes, the associated action runs to completion and then repeats.
This action is reversible; it creates a new action that is the reverse of the specified action and then repeats it forever.
Note
The action to be repeated must have a non-instantaneous duration.

13.3.18 reversedAction as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that reverses the behavior of another action.
Notes: This method always returns an action object; however, not all actions are reversible. When reversed, some actions return an object that either does nothing or performs the same action as the original action. For details on how an action is reversed, see the description of the class method used to create that action.

13.3.19 rotateBy(xAngle as double, yAngle as double, zAngle as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that rotates the node in each of the three principal axes by angles relative to its current orientation.
Notes: xAngle: The amount to rotate the node counterclockwise around the x-axis of its local coordinate space, in radians.
yAngle: The amount to rotate the node counterclockwise around the y-axis of its local coordinate space, in radians.
zAngle: The amount to rotate the node counterclockwise around the z-axis of its local coordinate space, in radians.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs rotation property animates to the new angle.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.rotateBy(-xAngle, -yAngle, -zAngle, duration)

13.3.20 rotateByAngle(angle as double, axis as SCNVector3MBS, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that rotates the node by an angle around a specified axis.
Notes: angle: The amount to rotate the node counterclockwise around the specified axis, in radians.
axis: A vector in the node,Äôs local coordinate space whose direction specifies the axis of rotation.
duration: The duration, in seconds, of the animation.

Return a new action object.

When the action executes, the node,Äôs rotation property animates to the new angle.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.rotateByAngle(-angle, axis, duration)

13.3.21 rotateTo(xAngle as double, yAngle as double, zAngle as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that rotates the node to absolute angles in each of the three principal axes.
Notes: xAngle: The amount to rotate the node counterclockwise around the x-axis of its local coordinate space, in radians.
yAngle: The amount to rotate the node counterclockwise around the y-axis of its local coordinate space, in radians.
zAngle: The amount to rotate the node counterclockwise around the z-axis of its local coordinate space, in radians.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs rotation property animates to the new angle. Calling this method is equivalent to calling rotateTo() and passing false for the shortestUnitArc parameter.
This action is not reversible; the reverse of this action has the same duration but does not change anything.
See also:

- 13.3.22 rotateTo(xAngle as double, yAngle as double, zAngle as double, duration as double, shortestUnitArc as boolean) as SCNActionMBS 873

13.3.22 rotateTo(xAngle as double, yAngle as double, zAngle as double, duration as double, shortestUnitArc as boolean) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that rotates the node to absolute angles in each of the three principal axes.
Notes: xAngle: The amount to rotate the node counterclockwise around the x-axis of its local coordinate space, in radians.
yAngle: The amount to rotate the node counterclockwise around the y-axis of its local coordinate space, in radians.
zAngle: The amount to rotate the node counterclockwise around the z-axis of its local coordinate space, in radians.
duration: The duration, in seconds, of the animation.

shortestUnitArc: If false (the default), the animation interpolates each component of the node,Äôs rotation between its current value and the new value. If true, the animation makes the most direct rotation possible from the node,Äôs current orientation to the new orientation.

Returns a new action object.

When the action executes, the node,Äôs rotation property animates to the new angle.
This action is not reversible; the reverse of this action has the same duration but does not change anything.
See also:

- 13.3.21 rotateTo(xAngle as double, yAngle as double, zAngle as double, duration as double) as SCNActionMBS 873

13.3.23 rotateToAxisAngle(axisAngle as SCNVector4MBS, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that rotates the node to an absolute angle around a specified axis.
Notes: axisAngle: A four-component vector whose first three components are a vector in the node,Äôs local coordinate space specifying an axis and whose fourth component is the amount to rotate the node counterclockwise around that axis, in radians.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs rotation property animates to the new angle.
This action is not reversible; the reverse of this action has the same duration but does not change anything.

13.3.24 runBlock(del as SCNActionRunBlockMBS, tag as variant = nil) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that executes run the given method.
Notes: When the action executes, SceneKit calls the delegate method. This action takes place instantaneously.
This action is not reversible; the reverse action executes the same delegate method.

13.3.25 scaleBy(scale as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that uniformly changes the scale factor of a node by a relative value.
Notes: scale: The amount of change to make to all three components of the node,Äôs scale.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs scale property animates to the new value.
This action is reversible; the reverse is created as if the following code had been executed:

SCNActionMBS.scaleBy(-scale, duration)

13.3.26 scaleTo(scale as double, duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that uniformly changes the scale factor of a node to an absolute value.
Notes: scale: The new value for all three components of the node,Äôs scale.
duration: The duration, in seconds, of the animation.

Returns a new action object.

When the action executes, the node,Äôs scale property animates to the new value.
This action is not reversible; the reverse of this action has the same duration but does not change anything.

13.3.27 sequence(actions() as SCNActionMBS) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that runs a collection of actions sequentially.
Notes: actions: An array of SCNActionMBS objects.
Returns a new sequence action object.

When the action executes, the first action in the sequence starts and runs to completion. Subsequent actions in the sequence run in a similar fashion until all of the actions in the sequence have executed. The duration of the sequence action is the sum of the durations of the actions in the sequence.

This action is reversible; it creates a new sequence action that reverses the order of the actions. Each action

in the reversed sequence is itself reversed.

13.3.28 unhide as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that ensures a node is not hidden.
Notes: When the action executes, the node,Äôs hidden property is set to false.
This action is reversible; the reverse is equivalent to the hide action.

13.3.29 wait(duration as double) as SCNActionMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an action that idles for a specified period of time.
Notes: When the action executes, the action waits for the specified amount of time and then ends. This is typically used as part of a sequence of actions to insert a delay between two other actions. You might also use it in conjunction with the runAction method to trigger code that needs to run at a later time.
This action is not reversible; the reverse of this action is the same action.

13.3.30 Properties

13.3.31 duration as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The duration required to complete an action.
Notes: This is the expected duration of an action,Äôs animation. The actual time an action takes to complete is modified by the action,Äôs timingMode property.
(Read and Write property)

13.3.32 Handle as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object references.
Notes: (Read and Write property)

13.3.33 speed as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A speed factor that modifies how fast an action runs.
Notes: The speed factor adjusts how fast an action,Äôs animation runs. For example, a speed factor of 2.0 means the animation runs twice as fast.
(Read and Write property)

13.3.34 timingMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The timing mode used to execute an action.
Notes: For possible values, see TimingMode constants. The default value is TimingModeLinear.
(Read and Write property)

13.3.35 Constants

Timing Modes

Constant	Value	Description
TimingModeEaseIn	1	Ease-in pacing. The animation begins slowly, and then speeds up as it progresses.
TimingModeEaseInEaseOut	3	Ease-in ease-out pacing. The animation begins slowly, accelerates through the middle of its duration, and then slows again before completing.
TimingModeEaseOut	2	Ease-out pacing. The animation begins quickly, and then slows as it completes.
TimingModeLinear	0	Linear pacing. The animation progresses evenly throughout its duration.

13.3.36 Delegates

13.3.37 SCNActionRunBlockMBS(node as SCNNodeMBS, tag as variant)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate for the runBlock method.

13.4 class SCNAudioPlayerMBS

13.4.1 class SCNAudioPlayerMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A controller for playback of a positional audio source in a SceneKit scene.
Example:

```
Dim file As FolderItem = SpecialFolder.Desktop.Child("Laser.wav")
audioSource = New SCNAudioSourceMBS(file)
audioPlayer = New MySCNAudioPlayerMBS(audioSource)
audioSource.Volume = 1.0
audioSource.Loops = true
audioSource.load

// now build your nodes
dim someNode as SCNNodeMBS

// show scene
MyView.scene = MyScene
MyView.play

// and add player to plays
someNode.addAudioPlayer(audioPlayer)
```

Notes: An SCNAudioPlayerMBS object controls playback of a positional audio source in a SceneKit scene. To use positional audio, first create a reusable SCNAudioSourceMBS or AVAudioNodeMBS object to provide an audio stream. Then, create an audio player to control the playback of that audio source. Finally, attach the audio player to an SCNNodeMBS object for spatialized 3D audio playback based on the position of that node relative to the scene,Äôs audioListener node.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3
- MBS Xojo Plugins, version 22.3pr3

13.4.2 Methods

13.4.3 audioPlayerWithAVAudioNode(audioNode as Variant) as SCNAudioPlayerMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns an audio player (creating one if necessary) to play an audio node.
Notes: Please pass a AVAudioNodeMBS class.

Creating an audio player from an audio node does not attach that node to an AVAudioEngine object for mixing and playback—instead, this option allows you to build your own audio node graph for custom effects before manually adding it to the audioEngine object of the view (or other SceneKit renderer) displaying the scene. Then, after you attach the audio player to an SCNNodeMBS object, SceneKit continually updates the audio node,Äôs 3D mixing parameters to match the position of the scene node.

When you use this method, SceneKit creates an audio player for the specified AVAudioNodeMBS object or, if an audio player for that node already exists and is not in use, SceneKit reuses the existing player object.

13.4.4 audioPlayerWithSource(audioSource as SCNAudioSourceMBS) as SCNAudioPlayerMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns an audio player (creating one if necessary) to play a simple audio source.
Notes: An SCNAudioSourceMBS object represents a distinct source of audio (for example, a sound file) that can be reused and shared by many player objects. Use this method to create an audio player from such a source. When you use this method, SceneKit creates an audio player for the specified SCNAudioSourceMBS object or, if an audio player for that source already exists and is not in use, SceneKit reuses the existing player object.

If instead you want to attach a more complex audio setup to a position in your scene—such as a mixer that varies several audio inputs in response to user input—build that setup using AVAudioNodeMBS objects and create a player for it with the audioPlayerWithAVAudioNode() method.

13.4.5 Constructor(audioSource as SCNAudioSourceMBS)

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Initializes an audio player for playing the specified simple audio source.
Notes: Using this initializer is typically not necessary. Instead, call the audioPlayerWithSource() method, which returns a cached audio player object if one for the specified audio source has already been created and is available for use.

13.4.6 Properties

13.4.7 audioNode as Variant

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The audio node SceneKit uses for mixing audio from this player.
Notes: Value is an AVAudioNodeMBS class.
(Read only property)

13.4.8 audioSource as SCNAudioSourceMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The source of audio played by this player.
Notes: An SCNAudioSourceMBS object represents a distinct source of audio—for example, a sound file—that can be reused and shared by many player objects. Use a player,Äôs audio source to configure the default values for playback parameters such as volume and reverb. To vary those parameters in real time during playback, use the audioNode property to work with the underlying AVAudioNodeMBS object.
If the player was created with the audioPlayerWithAVAudioNode() method, this property,Äôs value is nil.
(Read only property)

13.4.9 Handle as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.4.10 Events

13.4.11 DidFinishPlayback

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: .

Function: An event called by SceneKit when playback of the player,Äôs audio source has completed.
Notes: The block takes no parameters and returns no value. Use this block to perform actions when a sound finishes playing. For example, after a line of spoken character dialogue finishes playing, you might start playing another line of dialogue.

13.4.12 WillStartPlayback

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: .

Function: An event called by SceneKit when playback of the player,Äôs audio source is about to begin.

13.5 class SCNAudioSourceMBS

13.5.1 class SCNAudioSourceMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A simple, reusable audio source—music or sound effects loaded from a file—for use in positional audio playback.
Example:

```
Dim file As FolderItem = SpecialFolder.Desktop.Child("Laser.wav")
audioSource = New SCNAudioSourceMBS(file)
audioSource.Volume = 1.0
audioSource.Loops = true
audioSource.load

// now use for SCNAudioPlayerMBS class
```

Notes: To create positional audio effects, create an SCNAudioPlayerMBS object from the audio source to control playback, and add that player object to an SCNNodeMBS object in your scene. SceneKit then automatically spatializes 3D audio effects based on the position of that node relative to the scene,Äôs audioListener node.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3
- MBS Xojo Plugins, version 22.3pr3

13.5.2 Methods

13.5.3 audioSourceNamed(Name as String) as SCNAudioSourceMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns the audio source associated with the specified filename.
Notes: fileName: The name of an audio file. If this filename has not been previously requested, the method looks for an audio file with the specified name in the application,Äôs main bundle.

This method looks in the system caches for an audio source with the specified name and returns that object if it exists. If a matching audio source is not already in the cache, this method locates the audio file with the specified name in the application,Äôs main bundle, then creates a new audio source and caches it for reuse.

13.5.4 Constructor(File as FolderItem)

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Initializes an audio source from the specified audio file.
See also:

- 13.5.5 Constructor(URL as String) 883

13.5.5 Constructor(URL as String)

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Initializes an audio source from the specified audio file.
See also:

- 13.5.4 Constructor(File as FolderItem) 883

13.5.6 copy as SCNAudioSourceMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Creates a copy of the audio source object.

13.5.7 Load

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Loads audio data from the source and prepares it for playing.
Notes: This method reads audio data from the source file (specified when initializing the audio source) and performs any decompression necessary to prepare for playing audio. Use this method to control when your app or game incurs the run-time performance cost of such work—for example, you can load all audio source before starting a game level, instead of suffering a frame rate drop upon playing a new audio source during gameplay.
This method has no effect if the shouldStream property,Äôs value is true.

13.5.8 Properties

13.5.9 Handle as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.5.10 Loops as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the audio source should play repeatedly.
Notes: If this value is true, audio players using this source automatically begin playing again after playback has finished. If this value is false (the default), the audio source plays exactly once.
(Read and Write property)

13.5.11 Positional as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether audio from this source uses 3D positional mixing.
Notes: If this value is YES (the default), SceneKit mixes audio from the source based on its position relative to the scene,Äôs audioListener node—that is, the audio source,Äôs volume, reverb, and other parameters automatically change depending on the distance to the listener and other objects in the scene. (To position an audio source in a scene, create an SCNAudioPlayerMBS player from the source and attach that player to an SCNNodeMBS object.)
If you set this property to NO, the source,Äôs audio plays with the same volume (and other mixing parameters) regardless of the listener,Äôs position.
(Read and Write property)

13.5.12 Rate as Single

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The default playback rate for the audio source.
Notes: This property determines the default rate for when a source begins playing. To vary the rate during playback through an SCNAudioPlayerMBS object, use the player,Äôs audioNode property to access real-time audio controls.
(Read and Write property)

13.5.13 ReverbBlend as Single

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The default blend of blend of unmodified and reverb-processed (also called dry and wet) audio

for playback of the audio source.
Notes: This property determines the default reverb blend for when a source begins playing. To vary the reverb blend during playback through an SCNAudioPlayerMBS object, use the player,Äôs audioNode property to access real-time audio controls.
(Read and Write property)

13.5.14 ShouldStream as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the audio source should stream content from its source URL when playing.
Notes: If this value is true, audio players using this source do not preload audio buffer data, instead reading directly from the source file while playing audio. If this value is false, SceneKit loads audio buffer data upon playing audio from the source.
(Read and Write property)

13.5.15 Volume as Single

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The default playback volume for the audio source.
Notes: This property determines the default volume for when a source begins playing. To vary the volume during playback through an SCNAudioPlayerMBS object, use the player,Äôs audioNode property to access real-time audio controls.
(Read and Write property)

13.6 class SCNAvoidOccluderConstraintMBS

13.6.1 class SCNAvoidOccluderConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNAvoidOccluderConstraint constraints place the receiver at a position that prevent nodes with the specified category to occlude the target.
Notes: The target node and it's children are ignored as potential occluders.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.6.2 Methods

13.6.3 avoidOccluderConstraintWithTarget(target as SCNNodeMBS) as SCNAvoidOccluderConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNAvoidOccluderConstraint object.

13.6.4 Constructor(target as SCNNodeMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNAvoidOccluderConstraint object.

13.6.5 Properties

13.6.6 bias as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the bias the apply after moving the receiver to avoid occluders.
Notes: Defaults to 10e-5.
A positive bias will move the receiver closer to the target.
(Read and Write property)

13.6.7 occluderCategoryBitMask as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the category of node to consider as occluder.
Notes: Defaults to 1.
(Read and Write property)

13.6.8 target as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the target node.
Notes: (Read and Write property)

13.7 class SCNBillboardConstraintMBS

13.7.1 class SCNBillboardConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A constraint that orients a node to always point toward the current camera.
Notes: An SCNBillboardConstraint object automatically adjusts a node,Äôs orientation so that its local z-axis always points toward the pointOfView node currently being used to render the scene. For example, you can use a billboard constraint to efficiently render parts of a scene using two-dimensional sprite images instead of three-dimensional geometry—by mapping sprites onto planes affected by a billboard constraint, the sprites maintain their orientation with respect to the viewer. To attach constraints to an SCNNodeMBS object, use its constraints property.
Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.7.2 Methods

13.7.3 billboardConstraint as SCNBillboardConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNBillboardConstraint constraint.
Notes: A billboard constraint forces the receiver to look into the direction of the current point of view.

13.7.4 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNBillboardConstraint constraint.
Notes: A billboard constraint forces the receiver to look into the direction of the current point of view.

13.7.5 Properties

13.7.6 freeAxes as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the axes on which the billboarding orientation operates. Defaults to SCNBillboardAxisAll.

Notes: (Read and Write property)

13.7.7 Constants

Axes

Constant	Value	Description
SCNBillboardAxisAll	7	Align an affected node such that its orientation always matches that of the view. This is the default option for newly created billboard constraints.
SCNBillboardAxisX	1	Align an affected node such that its x-axis is always parallel to that of the view, leaving it free to rotate otherwise.
SCNBillboardAxisY	2	Align an affected node such that its y-axis is always parallel to that of the view, leaving it free to rotate otherwise.
SCNBillboardAxisZ	4	Align an affected node such that its z-axis is always perpendicular to the viewing plane, leaving it free to rotate otherwise.

13.8 class SCNBoxMBS

13.8.1 class SCNBoxMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNBox represents a box with rectangular sides and optional chamfers.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.4, page 41: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 36: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 22 to 24: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.8.2 Methods

13.8.3 box(width as double, height as double, length as double, chamferRadius as double) as SCNBoxMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a box with given width, height, length and chamfer radius.
Notes: width: The width of the box.
height: The height of the box.
length: The length of the box.
chamferRadius: The chamfer radius of the box.

13.8.4 Constructor(width as double, height as double, length as double, chamferRadius as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a box with given width, height, length and chamfer radius.
Notes: width: The width of the box.
height: The height of the box.
length: The length of the box.
chamferRadius: The chamfer radius of the box.

13.8.5 Properties

13.8.6 chamferRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The chamfer radius. Animatable.
Notes: If the value is strictly less than 0, the geometry is empty. The default value is 0.
(Read and Write property)

13.8.7 chamferSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of chamfer subdivisions. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 10.
(Read and Write property)

13.8.8 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the box. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.8.9 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.8.10 length as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The length of the box. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.

(Read and Write property)

13.8.11 lengthSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Z axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.8.12 width as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The width of the box. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.8.13 widthSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the X axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.9 class SCNCameraControllerMBS

13.9.1 class SCNCameraControllerMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The class for a camera controller.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 18.5pr3

13.9.2 Methods

13.9.3 clearRoll

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Clear the camera roll if any.

13.9.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.9.5 dollyToTarget(delta as double)

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Zoom by moving the camera along the axis from the camera position to the target.

13.9.6 rollAroundTarget(delta as double)

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Rotate the camera around the axis from the camera position to the target.
Notes: Delta is in degrees.

13.9.7 rotate(deltaX as double, deltaY as double)

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Rotate delta is in degrees.

13.9.8 stopInertia

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Stop current inertia.

13.9.9 translateInCameraSpace(x as double, y as double, z as double)

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Translate the camera along the local X/Y/Z axis.

13.9.10 Properties

13.9.11 automaticTarget as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Automatically update the target in beginInteraction.
Notes: Defaults to false.
(Read and Write property)

13.9.12 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.9.13 inertiaEnabled as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Set to true to enable inertia on endInteraction.
Notes: Defaults to false.
(Read and Write property)

13.9.14 inertiaFriction as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The friction coefficient applied to the inertia.
Notes: Defaults to 0.05.
(Read and Write property)

13.9.15 inertiaRunning as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns true if inertia is running.
Notes: (Read only property)

13.9.16 maximumHorizontalAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Minimum and maximum horizontal view angles in degrees for SCNInteractionModeFly and SCNInteractionModeOrbitTurntable.
Notes: The angle constraints is not enforced if both horizontal angle properties values are set to 0.
The angle constraints will not be enforced if the initial orientation is outside the given range.
The minimum angle must be inferior to the maximum angle.
Angles are in world space and within the range [-180, 180] .
Defaults to 0.0.
(Read and Write property)

13.9.17 maximumVerticalAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Minimum and maximum vertical view angles in degrees for SCNInteractionModeFly and SCNInteractionModeOrbitTurntable.
Notes: The angle constraints is not enforced if both vertical angle properties values are set to 0.
The angle constraints will not be enforced if the initial orientation is outside the given range.
The minimum angle must be inferior to the maximum angle.

Angles are in world space and within the range [-90, 90] .
Defaults to 0.0.
For example: set to minimum to 0 and maximum to 90 to only allow orbit around the top hemisphere.
(Read and Write property)

13.9.18 minimumHorizontalAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Minimum and maximum horizontal view angles in degrees for SCNInteractionModeFly and SCNInteractionModeOrbitTurntable.
Notes: The angle constraints is not enforced if both horizontal angle properties values are set to 0.
The angle constraints will not be enforced if the initial orientation is outside the given range.
The minimum angle must be inferior to the maximum angle.
Angles are in world space and within the range [-180, 180] .
Defaults to 0.0.
(Read and Write property)

13.9.19 minimumVerticalAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Minimum and maximum vertical view angles in degrees for SCNInteractionModeFly and SCNInteractionModeOrbitTurntable.
Notes: The angle constraints is not enforced if both vertical angle properties values are set to 0.
The angle constraints will not be enforced if the initial orientation is outside the given range.
The minimum angle must be inferior to the maximum angle.
Angles are in world space and within the range [-90, 90] .
Defaults to 0.0.
For example: set to minimum to 0 and maximum to 90 to only allow orbit around the top hemisphere.
(Read and Write property)

13.9.20 pointOfView as SCNNodeMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The node describing the point of view.
Notes: (Read and Write property)

13.9.21 target as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The camera target in world space for orbit rotation.
Notes: (Read and Write property)

13.9.22 worldUp as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The up vector in world space used as reference for SCNInteractionModeFly and SCNInteraction-ModeOrbitTurntable camera modes.
Notes: Defaults to (0, 1, 0).
(Read and Write property)

13.10 class SCNCameraMBS

13.10.1 class SCNCameraMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A set of camera attributes that can be attached to a node to provide a point of view for displaying the scene.
Notes: To display a scene, you must designate a node whose camera property contains a camera object as the point of view.

The SCNNode object containing a camera defines a point of view—that is, the position and orientation of the camera. A camera,Äôs direction of view is always along the negative z-axis of the node,Äôs local coordinate system. To point the camera at different parts of your scene, use the position, rotation, or transform property of the node containing it. (Alternatively, to ensure that a camera always points at a particular element of your scene even when that element moves, attach a SCNLookAtConstraint object to the node containing the camera.)

An SCNCamera object itself defines the shape and, in part, the appearance of the rendered scene as seen from its point of view. By default, a camera defines a perspective projection, whose field of view (FOV) and near and far visibility limits you control using the properties listed in Adjusting Camera Perspective and illustrated below.
Blog Entries

- MBS Xojo Plugins, version 18.5pr3

13.10.2 Methods

13.10.3 camera as SCNCameraMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new camera object.
Notes: To use the camera to display a scene, attach it to the camera property of a node and then select that node using the pointOfView property of the view (or layer or renderer) rendering the scene.

13.10.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new camera object.
Notes: To use the camera to display a scene, attach it to the camera property of a node and then select that node using the pointOfView property of the view (or layer or renderer) rendering the scene.

13.10.5 copy as SCNCameraMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the camera.

13.10.6 Properties

13.10.7 ApertureBladeCount as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The number of physical camera aperture blades simulated by SceneKit for depth-of-field effects.
Notes: When the wantsDepthOfField setting is enabled, SceneKit renders scenes using the camera with a depth-of-field blur (also called bokeh) effect modeled after those created by a real-world physical camera. One feature of real-world camera bokeh effects is the tendency of distant bright points to blur into larger shapes based on the shape of the aperture between the camera's lens and its imaging plane (film or sensor). Physical cameras control aperture using a mechanism that moves several flat blades in or out to create a smaller or larger opening, so the natural bokeh effect in traditional photography produces polygon-shaped blur effects.
This property controls the number of blades in the simulated camera aperture, and thus the polygon shape seen in the resulting bokeh effect. For example, a blade count of 6 (the default) causes distant bright points to blur into hexagon shapes. Increasingly large blade counts result in the bokeh effect appearing more circular, as shown below.
(Read and Write property)

13.10.8 AutomaticallyAdjustsZRange as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the camera automatically adjusts its zNear and zFar depth limits.
Notes: The default value of this property is false, specifying that the camera,Äôs zNear and zFar properties control its depth limits. If you change this property,Äôs value to true, SceneKit automatically adjusts the depth limits at render time to fit the bounding box of the scene. Changing the values of the zNear and zFar properties automatically resets this property,Äôs value to false.
(Read and Write property)

13.10.9 averageGray as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The luminance level to use as the midpoint of a tone mapping curve.

Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties, along with this property which serves as a constant estimate of scene luminance.
The default value is 0.18. By setting this property to a higher or lower value, you can compensate for scenes with darker or brighter content. Alternatively, by setting the wantsExposureAdaptation property, you can allow SceneKit to automatically adjust exposure as the visible contents of the scene change.
This property has no effect if the wantsHDR value is false. If the exposureAdaptationDarkeningSpeedFactor value is true, SceneKit ignores this property, and instead computes the average luminance currently visible to the camera during rendering.
(Read and Write property)

13.10.10 bloomBlurRadius as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The radius, in pixels, for the blurring portion of the bloom effect applied to highlights in the rendered scene. Animatable.
Notes: A bloom effect adds a soft glow to highlights (areas of bright color) in the rendered scene, simulating the way bright highlights appear to the human eye or a physical camera in a real-world scene. The bloom effect combines selective brightening and blurring effects; this property controls the blur portion of the effect. A value of zero effectively disables the bloom effect, and higher values result in a broader, softer glow. The default value is 4.0 pixels.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.11 bloomIntensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The magnitude of bloom effect to apply to highlights in the rendered scene. Animatable.
Notes: A bloom effect adds a soft glow to highlights (areas of bright color) in the rendered scene, simulating the way bright highlights appear to the human eye or a physical camera in a real-world scene. This property controls the strength of the bloom effect; lower values result in a subtle effect, and higher values create very bright glow. The default value is 0.0, resulting in no bloom effect.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.12 bloomThreshold as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The brightness threshold at which to apply a bloom effect to highlights in the rendered scene. Animatable.
Notes: A bloom effect adds a soft glow to highlights (areas of bright color) in the rendered scene, simulating the way bright highlights appear to the human eye or a physical camera in a real-world scene. This property controls the brightness level required to trigger the bloom effect; lower values apply the effect to more of the scene, and higher values apply the effect only to the brightest white areas. The default value is 1.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.13 colorFringeIntensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The blend factor for fading the color fringing effect applied to the rendered scene.
Notes: Color fringing applies an effect that separately blurs the color components of each rendered pixel, adding subtle rainbow edge effects to the rendered scene that simulate the effects of chromatic aberration in a physical camera. Higher values for this property result in brighter, more vivid color fringing, and lower values create a subtler effect. The default value of 1.0 leaves the color fringing effect at its most vivid.
This property controls a fade between the color fringing effect and the otherwise-normally-rendered image. The colorFringeStrength property controls the breadth of the color fringing effect.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.14 colorFringeStrength as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The magnitude of color fringing effect to apply to the rendered scene.
Notes: Color fringing applies an effect that separately blurs the color components of each rendered pixel, adding subtle rainbow edge effects to the rendered scene that simulate the effects of chromatic aberration in a physical camera. Higher values create a more pronounced color shift, creating wider rainbow fringes; lower values spread colors across shorter distances, creating a subtler effect. The default value of 0.0 disables the color fringing effect entirely.
This property controls the breadth of color fringing. The colorFringeIntensity property controls the blend factor between the color-fringed and the otherwise-normally-rendered image.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.15 colorGrading as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A texture for applying color grading effects to the entire rendered scene.
Notes: The contents value for this material property must be a 3D color lookup table, or a 2D texture image that represents such a table arranged in a horizontal strip.

see also
https://developer.apple.com/documentation/scenekit/scncamera/1644114-colorgrading
(Read only property)

13.10.16 contrast as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: An adjustment factor to apply to the overall visual contrast of the rendered scene.
Notes: A value of 0.0 (the default) leaves the rendered scene unchanged. Positive values increase contrast between bright and dark areas, and negative values reduce contrast, shifting the rendered scene towards a uniform gray.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.17 exposureAdaptationBrighteningSpeedFactor as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The relative duration of automatically animated exposure transitions from dark to bright areas.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. When the wantsExposureAdaptation property is enabled, SceneKit automatically adjusts the tone mapping curve based on the average luminance currently visible to the camera, and creates automatic transitions between exposure levels.
SceneKit automatically determines the overall duration of exposure-level animations based on the values of this property and the exposureAdaptationDarkeningSpeedFactor property. The default value is 0.4, resulting in brightening animations that are slightly faster than darkening animations.
This property has no effect if either of the wantsHDR or wantsExposureAdaptation values is false.
(Read and Write property)

13.10.18 exposureAdaptationDarkeningSpeedFactor as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The relative duration of automatically animated exposure transitions from bright to dark areas.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. When the wantsExposureAdaptation property is enabled, SceneKit automatically adjusts the tone mapping curve based on the average luminance currently visible to the camera, and creates automatic transitions between exposure levels.
SceneKit automatically determines the overall duration of exposure-level animations based on the values of this property and the exposureAdaptationDarkeningSpeedFactor property. The default value is 0.6, resulting in darkening animations that are slightly faster than brighting animations.
This property has no effect if either of the wantsHDR or wantsExposureAdaptation values is false.
(Read and Write property)

13.10.19 exposureOffset as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A logarithmic bias that adjusts the results of SceneKit,Äôs tone mapping operation, brightening or darkening the visible scene.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties, along with a measure of scene luminance.
Use this property to bias the tone mapping curve. The default exposure offset is zero, specifying no bias. Positive values result in a brighter scene, and negative values result in a darker scene.
This property has no effect if the wantsHDR value is false.
(Read and Write property)

13.10.20 FieldOfView as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The vertical or horizontal viewing angle of the camera.
Notes: The projectionDirection property determines whether this fieldOfView property measures the camera's vertical or horizontal viewing angle, and SceneKit automatically calculates the viewing angle in the other direction to match the aspect ratio of the view displaying the scene. For example, a fieldOfView of 60 and the default SCNCameraProjectionDirectionVertical projection, presented fullscreen on a 16:9 display in portrait orientation, results in a vertical viewing angle of 60¬∞ and a horizontal viewing angle of 33.75¬∞.
You can choose to specify viewing angle either directly, using this fieldOfView property, or in terms that model a physical camera, using the sensorHeight and focalLength properties. Setting the fieldOfView property causes SceneKit to automatically recalculate the focalLength value, and setting the sensorHeight or focalLength property recalculates fieldOfView.
(Read and Write property)

13.10.21 FocalBlurSampleCount as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The number of pixel samples SceneKit uses to create depth-of-field blur effects.
Notes: When the wantsDepthOfField setting is enabled, SceneKit renders depth-of-field blur (also called bokeh) effects using a blur filter that samples multiple points in the image. Sampling a larger number of points produces a higher quality visual effect at a higher performance cost, and vice versa. The default sample count is 25.
(Read and Write property)

13.10.22 FocalLength as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The camera's focal length, in millimeters.
Notes: The sensorHeight and focalLength properties determine the camera's horizontal and vertical viewing angles using terms that model physical camera devices. (Alternatively, you can work with viewing angle directly though the fieldOfView property.) For example, with the default sensor height of 24 mm and default focal length of 50 mm, the vertical field of view is 60¬∞.
Setting the fieldOfView property causes SceneKit to automatically recalculate the focalLength value, and setting the sensorHeight or focalLength property recalculates fieldOfView.
(Read and Write property)

13.10.23 focusDistance as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The distance from the camera at which objects appear in sharp focus. Animatable.
Notes: Objects at this distance from the camera appear perfectly focused. Objects nearer to or farther from the camera than this distance appear increasingly blurred, with the behavior of the blur effect depending on the fStop, apertureBladeCount and focalBlurSampleCount properties. The default focus distance is 2.5.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.10.24 fStop as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The physical camera aperture simulated by SceneKit for depth-of-field effects. Animatable.
Notes: F-stop numbers describe the light-gathering area of a physical camera's imaging system, and are typically expressed as the denominator of a ratio including the camera's focal length í, such as í/2 or í/5.6. A larger denominator indicates a smaller aperture, allowing less light to pass from the camera's lens through

to the imaging plane (sensor or film), and a smaller denominator indicates a larger aperture that lets more light through.
SceneKit uses aperture measurements to simulate depth-of-field blur effects (also called bokeh) approximating those produced by a physical camera. A larger fStop number (or aperture denominator) causes most of the scene to appear in focus, with extremely close or far depths showing slight blurring; a smaller number results in only a narrow range of depths appearing in focus, and a more pronounced blur effect for the rest of the scene. The default fStop value is 5.6.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.10.25 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.10.26 maximumExposure as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum exposure value to use in tone mapping.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties, along with a measure of scene luminance.
Exposure values are exponential: a value of 1.0 doubles brightness, a value of 2.0 quadruples brightness, a value of -1.0 halves brightness, and so on. The default value is 15.0. Decreasing the value causes brighter portions of the scene to become over-exposed (uniformly white, losing definition). Increasing the value adds more dynamic range for brighter portions of the scene; however, a greater breadth of difference between the minimum and maximum exposures decreases contrast.
This property has no effect if the wantsHDR value is false.
(Read and Write property)

13.10.27 minimumExposure as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum exposure value to use in tone mapping.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties, along with a measure of scene luminance.

Exposure values are exponential: a value of 1.0 doubles brightness, a value of 2.0 quadruples brightness, a value of -1.0 halves brightness, and so on. The default value is -15.0. Increasing the value causes darker portions of the scene to become under-exposed (uniformly black, losing definition). Decreasing the value adds more dynamic range for darker portions of the scene; however, a greater breadth of difference between the minimum and maximum exposures decreases contrast.
This property has no effect if the wantsHDR value is false.
(Read and Write property)

13.10.28 motionBlurIntensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A factor that determines the intensity of motion blur effects. Animatable.
Notes: The default intensity of zero results in no motion blur effect. Higher values (toward a maximum of 1.0) create more pronounced motion blur effects.
Motion blur is not supported when wide-gamut color rendering is enabled. Wide-gamut rendering is enabled by default on supported devices; to opt out, set the SCNDisableWideGamut key in your app's Info.plist file.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.10.29 Name as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A name associated with the camera object.
Notes: You can provide a descriptive name for a camera object to make managing your scene graph easier. Cameras loaded from a scene file may have names assigned by an artist using a 3D authoring tool. Use the SCNSceneSourceMBS class to examine cameras in a scene file without loading its scene graph.
(Read and Write property)

13.10.30 orthographicScale as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the camera,Äôs magnification factor when using an orthographic projection.
Notes: In an orthographic projection, equally sized objects appear equally sized regardless of their distance from the camera. To switch between orthographic and perspective projections, see the usesOrthographicProjection property.
(Read and Write property)

13.10.31 ProjectionDirection as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The axis used to determine field of view or orthographic scale.
Notes: The fieldOfView property measures view angle in a single primary direction, determined by this projectionDirection property. For the other direction, SceneKit automatically adjusts field of view depending on the aspect ratio of the view presenting the scene.
For example, with the default projection direction of SCNCameraProjectionDirectionVertical, setting fieldOfView to 60 results in a vertical view angle of 60¬∞. If the scene appears on a display with a 4:3 aspect ratio, the horizontal view angle is 80¬∞. However, if the scene appears on a 16:9 display, the horizontal view angle is 106¬∞.
This property has a similar effect on scaling for orthographic projections. The orthographicScale property measures the scale factor in the direction of the projectionDirection property, and SceneKit automatically calculates scale factor in the other direction according to aspect ratio.
(Read and Write property)

13.10.32 projectionTransform as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The camera,Äôs projection transformation.
Notes: This transformation expresses the combination of all the camera,Äôs geometric properties: projection type (perspective or orthographic), field of view, depth limits, and orthographic scale (if applicable). SceneKit uses this transformation to convert points in the camera node,Äôs coordinate space to the renderer,Äôs 2D space when rendering and processing events.
You can use this transformation directly if your app needs to convert between view and renderer coordinates for other purposes. Alternatively, if you compute your own projection transform matrix, you can set this property to override the transformation synthesized from the camera,Äôs geometric properties.
(Read and Write property)

13.10.33 saturation as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: An adjustment factor to apply to the overall color saturation of the rendered scene.
Notes: A value of 1.0 (the default) leaves scene colors unchanged. Greater values result in oversaturated colors, and a value of 0.0 makes the rendered scene entirely grayscale.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.34 screenSpaceAmbientOcclusionBias as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: An offset for modulating ambient occlusion effects.
Notes: Ambient occlusion is an effect that improves material shading by calculating the amounts of ambient light that reach various parts of a surface, creating shadows on parts of a geometry where incoming light is obscured by other parts of the geometry. (You can provide pre-rendered ambient occlusion effects for a material using its ambientOcclusion property.) Screen-space ambient occlusion (SSAO) provides a real-time approximation of this effect for the entire scene viewed through the camera.
This screenSpaceAmbientOcclusionBias value is used in an intermediate stage of calculating the SSAO effect, and measures a distance in scene units. Increasing or decreasing this value from its default of 0.03 can help to offset unrealistic effects produced by changing other SSAO settings.
(Read and Write property)

13.10.35 screenSpaceAmbientOcclusionDepthThreshold as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum depth difference, in units of scene space, at which to apply ambient occlusion effects.
Notes: Ambient occlusion is an effect that improves material shading by calculating the amounts of ambient light that reach various parts of a surface, creating shadows on parts of a geometry where incoming light is obscured by other parts of the geometry. (You can provide pre-rendered ambient occlusion effects for a material using its ambientOcclusion property.) Screen-space ambient occlusion (SSAO) provides a real-time approximation of this effect for the entire scene viewed through the camera.
This screenSpaceAmbientOcclusionDepthThreshold property controls the effect of relative distance from the camera on SSAO effects. Higher values create more shadowing effects between foreground and background elements of the scene, but this can result in unrealistic dark halos around foreground elements that are far from the background. Lower values avoid dark halo effects, but create less visual separation between scene elements at different distances from the camera. The default value is 0.2 units.
(Read and Write property)

13.10.36 screenSpaceAmbientOcclusionIntensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The intensity of the screen-space ambient occlusion effect applied in camera rendering.
Notes: Ambient occlusion is an effect that improves material shading by calculating the amounts of ambient light that reach various parts of a surface, creating shadows on parts of a geometry where incoming light is obscured by other parts of the geometry. (You can provide pre-rendered ambient occlusion effects for a material using its ambientOcclusion property.) Screen-space ambient occlusion (SSAO) provides a real-time approximation of this effect for the entire scene viewed through the camera.
The default value of this property is zero, disabling SSAO effects. Increasing the intensity value creates deeper, bolder shadows.

(Read and Write property)

13.10.37 screenSpaceAmbientOcclusionNormalThreshold as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The magnitude of the blur effect applied to create ambient occlusion shadows.
Notes: Ambient occlusion is an effect that improves material shading by calculating the amounts of ambient light that reach various parts of a surface, creating shadows on parts of a geometry where incoming light is obscured by other parts of the geometry. (You can provide pre-rendered ambient occlusion effects for a material using its ambientOcclusion property.) Screen-space ambient occlusion (SSAO) provides a real-time approximation of this effect for the entire scene viewed through the camera.
SSAO shadowing includes a blur effect to realistically soften differences in shadow between adjacent pixels, which depends on both the smoothness of scene geometry and this factor. Larger blur factors create a softer, more spread-out blur; smaller factors create coarser shadowing effects.
(Read and Write property)

13.10.38 screenSpaceAmbientOcclusionRadius as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The distance, in units of scene space, at which ambient occlusion takes effect.
Notes: Ambient occlusion is an effect that improves material shading by calculating the amounts of ambient light that reach various parts of a surface, creating shadows on parts of a geometry where incoming light is obscured by other parts of the geometry. (You can provide pre-rendered ambient occlusion effects for a material using its ambientOcclusion property.) Screen-space ambient occlusion (SSAO) provides a real-time approximation of this effect for the entire scene viewed through the camera.
SSAO effects work by storing relevant scene geometry information for each pixel, and using that information to produce per-pixel shading effects. This screenSpaceAmbientOcclusionRadius property determines the area in scene space to consider around each pixel for determining the amount of incoming ambient light blocked by surrounding geometry (and thus the amount of shadow effect to apply). The default value is 5; smaller values cause SSAO effects to apply only to finer geometry details, while larger values affect coarser details.
(Read and Write property)

13.10.39 SensorHeight as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The vertical size of the camera's imaging plane, in millimeters.
Notes: The sensorHeight and focalLength properties determine the camera's horizontal and vertical viewing angles using terms that model physical camera devices. (Alternatively, you can work with viewing angle directly though the fieldOfView property.) For example, with the default sensor height of 24 mm and default

focal length of 50 mm, the vertical field of view is 60¬∞.
Setting the fieldOfView property causes SceneKit to automatically recalculate the focalLength value, and setting the sensorHeight or focalLength property recalculates fieldOfView.
(Read and Write property)

13.10.40 UsesOrthographicProjection as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the camera uses an orthographic projection.
Notes: The default value of this property is false, specifying a perspective projection. In a perspective projection, equally sized objects nearer to the camera appear larger than those farther away.
Set the value of this property to true to specify an orthographic projection. In an orthographic projection, equally sized objects appear equally sized regardless of distance from the camera.
To control the magnification factor of an orthographic camera, use its orthographicScale property.
(Read and Write property)

13.10.41 vignettingIntensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The magnitude of vignette (darkening around edges) effect to apply to the rendered scene.
Notes: A vignette effect darkens the edges and corners of the rendered scene, simulating the effect of lens and barrel shape on the image produced by a physical camera. Higher values result in more darkening, and lower values result in a subtler effect. The default value of 0.0 results in no vignetting effect.
This property controls the level of darkening applied; the vignettingPower property controls the area of the rendered image to be darkened.
To enable this behavior, you must first enable the wantsHDR setting.
(Read and Write property)

13.10.42 vignettingPower as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The amount of the rendered scene to darken with a vignette effect.
Notes: A vignette effect darkens the edges and corners of the rendered scene, simulating the effect of lens and barrel shape on the image produced by a physical camera. Higher values result apply the darkening effect to a broader area around the edges of the rendered image, and lower values apply the effect to a smaller area, leaving more of the rendered image at full brightness. The default value of 0.0 results in no vignetting effect.
This property controls the area of the rendered image to be darkened; the vignettingIntensity property controls the level of darkening applied to those areas.
To enable this behavior, you must first enable the wantsHDR setting.

(Read and Write property)

13.10.43 WantsDepthOfField as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit renders depth-of-field blur effects for the camera.
Notes: This value is false by default, disabling depth-of-field effects.
Enabling this property causes SceneKit to render blur effects that model those created by a physical camera device (also known as bokeh). That is, objects in the scene appear more or less blurry depending on their distance from the camera and the camera's focusDistance, and the intensity and style of the blur effect depend on the fStop and apertureBladeCount properties.

Note
For best results, also enable the wantsHDR property when using depth-of-field effects. High Dynamic Range rendering provides high contrast for distant bright points in the scene, creating more pronounced bokeh effects.
(Read and Write property)

13.10.44 WantsExposureAdaptation as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit automatically adjusts the exposure level.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. One measure of tone mapping is the exposure value, whose effect on the output is similar to that of the shutter speed (or exposure time) of a real-world camera—lower exposure values result in a darker image, and higher exposures result in a brighter image. You cannot adjust exposure value directly—instead, SceneKit determines a tone mapping curve (including the exposure level) from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties along with a measure of scene luminance.

If this property,Äôs value is true (the default), SceneKit automatically measures the current luminance visible to the camera during rendering, and adjusts the exposure level accordingly. Additionally, when the scene luminance changes, SceneKit automatically animates a transition to the new exposure level (see the exposureAdaptationBrighteningSpeedFactor and exposureAdaptationDarkeningSpeedFactor properties).

Note
The visual effect of automatic exposure is similar to how human visual perception adjusts to changes in environmental lighting. For example, consider a game scene where the player moves from a darkened area into full daylight. At first, the exposure value is low, allowing for visible detail in the darkened area, but

no detail in the white daylight outside. As the player moves into the daylight, the entire view becomes blindingly bright, but over a brief time the player,Äôs vision adapts: detail becomes visible in the bright area, and the darkened area loses detail.

If this property,Äôs value is false, SceneKit,Äôs tone mapping effect is constant. Instead of responding to scene luminance, SceneKit uses the averageGray property to determine the tone mapping curve.
This property has no effect if the wantsHDR value is false.
(Read and Write property)

13.10.45 WantsHDR as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit applies High Dynamic Range (HDR) post-processing effects to a scene.
Notes: When this property,Äôs value is false (the default), SceneKit performs lighting calculations in a color space whose brightness range is similar to that of the output display. This approach limits the ability to perform realistic rendering of scenes with fine details in brightness levels.
When you enable HDR rendering for a camera, SceneKit calculates lighting in a much deeper color space, preserving fine details in contrast regardless of brightness, then applies a post-processing effect called tone mapping to translate luminance values from that space to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve (including the exposure level) from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties along with a measure of scene luminance. The wantsExposureAdaptation property determines whether tone mapping effects are static or dynamically respond when the luminance visible to the camera changes.
The default value is false.
(Read and Write property)

13.10.46 whitePoint as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The luminance level to use as the upper end of a tone mapping curve.
Notes: When using a High Dynamic Range (HDR) camera, SceneKit applies a process called tone mapping to translate the wide range of luminance values in the visible scene to the narrower range of brightness values that can be shown on a display. SceneKit determines a tone mapping curve from the minimumExposure, maximumExposure, exposureOffset, and whitePoint properties, along with a measure of scene luminance.
The default value is 1.0. By setting this property to a higher or lower value, you can produce more gradual or more abrupt transitions between shadows and highlights.
This property has no effect if the wantsHDR value is false.
(Read and Write property)

13.10.47 zFar as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The camera,Äôs far depth limit. Animatable.
Notes: The far value determines the maximal distance between the camera and a visible surface. If a surface is farther from the camera than this distance, the surface is clipped and does not appear. The default far value is 100.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.10.48 zNear as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The camera's near depth limit. Animatable.
Notes: The near value determines the minimal distance between the camera and a visible surface. If a surface is closer to the camera than this distance, the surface is clipped and does not appear. The near value must not be zero. The default near value is 1.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.10.49 Constants

Orthographic Projection

Constant	Value	Description
kProjectionDirectionHorizontal	1	The camera's field of view or orthographic scale are measured horizontally.
kProjectionDirectionVertical	0	The camera's field of view or orthographic scale are measured vertically.

13.11 class SCNCapsuleMBS

13.11.1 class SCNCapsuleMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNCapsule represents a capsule with controllable height and cap radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.2, page 41: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 24 to 25: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.11.2 Methods

13.11.3 capsule(capRadius as double, height as double) as SCNCapsuleMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a capsule with given radius and height.
Notes: capRadius: The radius of the capsule.
height: The height of the capsule.

13.11.4 Constructor(capRadius as double, height as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a capsule with given radius and height.
Notes: capRadius: The radius of the capsule.
height: The height of the capsule.

13.11.5 Properties

13.11.6 capRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The cap radius of the capsule. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 0.5.
(Read and Write property)

13.11.7 capSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions in the cap. Animatable.
Notes: If the value is less than 2, the behavior is undefined. The default value is 24.
(Read and Write property)

13.11.8 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the capsule. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 2.
(Read and Write property)

13.11.9 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.11.10 radialSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the radial coordinate. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 48.

(Read and Write property)

13.12 class SCNConeMBS

13.12.1 class SCNConeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNCone represents a cone with controllable height, top radius and bottom radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 19.6, page 47: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.2, page 37: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 25 to 26: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.12.2 Methods

13.12.3 cone(topRadius as double, bottomRadius as double, height as double) as SCNConeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a cone with given top radius, bottom radius and height.
Notes: topRadius: The radius at the top of the cone.
bottomRadius: The radius at the bottom of the cone.
height: The height of the cone.

13.12.4 Constructor(topRadius as double, bottomRadius as double, height as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a cone with given top radius, bottom radius and height.
Notes: topRadius: The radius at the top of the cone.

bottomRadius: The radius at the bottom of the cone.
height: The height of the cone.

13.12.5 Properties

13.12.6 bottomRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The radius at the bottom of the cone. Animatable.
Notes: If the value is less than 0, the geometry is empty. The default value is 0.5.
(Read and Write property)

13.12.7 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the cone. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.12.8 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.12.9 radialSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the radial coordinate. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 48.
(Read and Write property)

13.12.10 topRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The radius at the top of the cone. Animatable.
Notes: If the value is less than 0, the geometry is empty. The default value is 0.
(Read and Write property)

13.13 class SCNConstraintMBS

13.13.1 class SCNConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The abstract superclass for objects that automatically adjust the position, rotation, or scale of a node based on specified rules.
Notes: To control the transform (position, rotation, and scale) of one or more SCNNodeMBS objects with constraints, create and configure instances of the SCNConstraint subclass that provides the behavior you want, then add those constraint objects to each node's constraints array.

When SceneKit prepares to render a scene, it examines the list of constraints attached to each node to determine the transform for that node, then applies the new transformation before displaying the scene.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.13.2 Methods

13.13.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

13.13.4 copy as SCNConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the constraint object.

13.13.5 Properties

13.13.6 Enabled as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether the constraint is enabled or not.
Notes: Defaults to true.
(Read and Write property)

13.13.7 Handle as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.13.8 Incremental as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether or not the contraint should applies incrementally and have it's effect being cumulated over the rendered frames.
Notes: Defaults to true on macOS 10.13, iOS 11, tvOS 11 and watchOS 4. Defaults to false in earlier versions.
(Read and Write property)

13.13.9 influenceFactor as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The influence of the constraint on the node,Äôs transformation.
Notes: Use this property to relax the effect of a constraint on the nodes it applies to. For example, consider a node containing a spotlight, constrained by an SCNLookAtConstraintMBS object to point toward another node containing a moving game character. If the constraint,Äôs influence factor is 1.0, SceneKit adjusts the spotlight node to point directly at the game character each time it renders a frame. If you reduce the influence factor to 0.5, each time SceneKit renders a frame it moves the spotlight halfway from its current orientation to the target orientation. As a result, the spotlight continues to follow the moving character, but with a slight lag.

The default influence factor is 1.0, specifying that SceneKit apply the full effect of the constraint every frame. An influence factor of 0.0 means the constraint has no effect.

This property has no effect on SCNTransformConstraintMBS objects.
(Read and Write property)

13.14 control SCNControlMBS

13.14.1 control SCNControlMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a SCNViewMBS.
Notes: This control embeds a special SCNView subclass.
Designed for Xojo 2013r1 and newer. May work on Xojo 2012, but not perfectly.
Please use view property to access the underlaying object and set properties.

To learn about rendering events, please check here:
https://developer.apple.com/documentation/scenekit/scnscenerendererdelegate?language=objc

You can set PreferLowPowerDevice and PreferredRenderingAPI properties before initializing a view.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- Three new controls for iOS in Xojo
- MBS Xojo Plugins, version 20.6pr3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.2
- MBS Xojo Plugins, version 20.2pr1
- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 19.6, page 45: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, page 39: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 26: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.14.2 Properties

13.14.3 View as SCNViewMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: The view used in the control.
Notes: Use this object to set more options on the control.
(Read only property)

13.14.4 Events

13.14.5 BoundsChanged

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

13.14.6 Close

Plugin Version: 18.4, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

13.14.7 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

13.14.8 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: Called when a menuitem is choosen.
Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

13.14.9 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

13.14.10 EnableMenuItems

Plugin Version: 18.4, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

13.14.11 FrameChanged

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

13.14.12 GotFocus

Plugin Version: 18.4, Platform: macOS, Targets: .

Function:

The control itself got focus.
In Xojo version 2021r3 and newer this event is named FocusReceived.
Notes:

This only fires if the control itself got focus and not a sub control.

13.14.13 LostFocus

Plugin Version: 18.4, Platform: macOS, Targets: .

Function:

The control lost focus.
In Xojo version 2021r3 and newer this event is named FocusLost.
Notes:

This only fires if the control itself lost focus and not a sub control.

13.14.14 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown.
If you return False, the system handles the MouseDown so the above event handlers do not get called.

13.14.15 MouseDrag(x as Integer, y as Integer)

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

13.14.16 MouseUp(x As Integer, y As Integer)

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

13.14.17 Open

Plugin Version: 18.4, Platform: macOS, Targets: .

Function:

The control is about to was created and you can initialize it.
In Xojo version 2021r3 and newer this event is named Opening.

13.14.18 rendererDidApplyAnimations(time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur after actions and animations are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this method to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.14.19 rendererDidApplyConstraints(time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Invoked on the control once the scene renderer did apply the constraints.
Notes: time The time at which the constraints were simulated.

All modifications done within this method don't go through the transaction model, they are directly applied on the presentation tree.

13.14.20 rendererDidRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the control that the renderer has rendered the scene.
Notes: scene: The scene object that was rendered.
time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this method to perform custom drawing after SceneKit has rendered a scene—for example, to draw overlay content on top of SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this method—the results of modifying SceneKit objects in this method are undefined.

- To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
- To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this method. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.14.21 rendererDidSimulatePhysics(time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur after physics simulations are performed.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this event exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

This event is the last opportunity SceneKit provides for you to change the scene graph before rendering.

13.14.22 rendererUpdate(time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the control to perform any updates that need to occur before actions, animations, and physics are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.14.23 rendererWillRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the control that the renderer has cleared the viewport and is about to render the scene.
Notes: scene: The SCNSceneMBS object to be rendered.
time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this event to perform custom drawing before SceneKit renders a scene—for example, to draw backdrop content underneath SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this event—the results of modifying SceneKit objects during this event are undefined.

To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this event. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.14.24 ScaleFactorChanged(NewFactor as double)

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

13.14.25 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 24.1, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

13.15 class SCNCylinderMBS

13.15.1 class SCNCylinderMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNCylinder represents a cylinder with controllable height and radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 19.6, page 47: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, page 42: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 37: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 27: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.15.2 Methods

13.15.3 Constructor(radius as double, height as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a cylinder with given radius and height.

13.15.4 cylinder(radius as double, height as double) as SCNCylinderMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a cylinder with given radius and height.

13.15.5 Properties

13.15.6 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the cylinder. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.15.7 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.15.8 radialSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the radial coordinate. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 48.
(Read and Write property)

13.15.9 radius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The radius of the cylinder. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 0.5.
(Read and Write property)

13.16 class SCNDistanceConstraintMBS

13.16.1 class SCNDistanceConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNDistanceConstraint ensure a minimum/maximum distance with a target node.
Notes: Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.16.2 Methods

13.16.3 Constructor(target as SCNNodeMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNDistanceConstraint constraint.

13.16.4 distanceConstraintWithTarget(target as SCNNodeMBS) as SCNDistanceConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNDistanceConstraint constraint.

13.16.5 Properties

13.16.6 maximumDistance as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum distance.
Notes: Defaults to MAXFLOAT. Animatable.
(Read and Write property)

13.16.7 minimumDistance as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum distance.
Notes: Defaults to 0. Animatable.
(Read and Write property)

13.16.8 target as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the target node to keep distance with.
Notes: (Read and Write property)

13.17 class SCNFloorMBS

13.17.1 class SCNFloorMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNFloor represents an infinite plane geometry.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

13.17.2 Methods

13.17.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a floor.
Notes: A floor is an infinite plane.

13.17.4 floor as SCNFloorMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a floor.
Notes: A floor is an infinite plane.

13.17.5 Properties

13.17.6 length as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The floor extent along the Z axis. Animatable.
Notes: If the value is equal to 0, the floor is infinite on the Z axis. The default value is 0.
(Read and Write property)

13.17.7 reflectionCategoryBitMask as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines the node categories to reflect. Defaults to all bits set.
Notes: (Read and Write property)

13.17.8 reflectionFalloffEnd as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the distance from the floor where the falloff finishes. Animatable.
Notes: If the value is 0 then there is no falloff. The default value is 0.
(Read and Write property)

13.17.9 reflectionFalloffStart as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the distance from the floor where the falloff begins. Animatable.
Notes: The default value is 0.
(Read and Write property)

13.17.10 reflectionResolutionScaleFactor as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the resolution scale factor of the buffer used to render the reflection.
Notes: Defaults to 1.0.
(Read and Write property)

13.17.11 reflectivity as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the reflectivity of the floor. Animatable.
Notes: If the value is greater than zero then the surface will reflect other objects in the scene. The default value is 0.25.
(Read and Write property)

13.17.12 width as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The floor extent along the X axis. Animatable.
Notes: If the value is equal to 0, the floor is infinite on the X axis. The default value is 0.
(Read and Write property)

13.18 class SCNGeometryElementMBS

13.18.1 class SCNGeometryElementMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: A container for index data describing how vertices connect to define a three-dimensional object, or geometry.
Notes: You use geometry elements together with SCNGeometrySourceMBS objects to define custom SCNGeometryMBS objects or to inspect the data that composes an existing geometry. You create a custom geometry using a three-step process:

1. Create one or more SCNGeometrySource objects, each of which defines per-vertex information such as position, surface normal, or texture coordinates for all vertices in the geometry.
2. Create at least one SCNGeometryElement object, containing an array of indices identifying vertices in the geometry sources and describing the drawing primitive that SceneKit uses to connect the vertices when rendering the geometry.
3. Create an SCNGeometry instance from the geometry sources and geometry elements.

When SceneKit renders a geometry, each geometry element corresponds to a drawing command sent to the GPU. Because different rendering states require separate drawing commands, you can define a geometry using multiple geometry elements. For example, the teapot geometry shown below has four geometry elements, so you can assign up to four SCNMaterial objects in order to render each element with a different appearance. But because each drawing command incurs a CPU time overhead when rendering, minimizing the number of elements in a custom geometry can improve rendering performance.
Blog Entries

- MBS Xojo Plugins, version 19.3pr5

13.18.2 Methods

13.18.3 Constructor(data as MemoryBlock, PrimitiveType as Integer, primitiveCount as Integer, bytesPerIndex as Integer)

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry element from the specified data and options.
Notes: data: The data describing the element.
primitiveType: The drawing primitive that connects vertices when rendering the geometry element. For possible values, see SCNGeometryPrimitiveType.
primitiveCount: The number of primitives in the element.
bytesPerIndex: The number of bytes that represent a single index value in the data.

Returns a new geometry element object.

An element,Äôs data is an array of index values identifying vertices in a geometry source. SceneKit interprets the data as an array of unsigned integers (whose size is specified by the bytesPerIndex parameter), and then connects the vertices in the order specified by this array, arranged according to the primitiveType parameter. To create a custom SCNGeometryMBS object from the geometry element, use the geometryWithSources() method.

13.18.4 geometryElementWithData(data as MemoryBlock, PrimitiveType as Integer, primitiveCount as Integer, bytesPerIndex as Integer) as SCNGeometryElementMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry element from the specified data and options.
Notes: data: The data describing the element.
primitiveType: The drawing primitive that connects vertices when rendering the geometry element. For possible values, see SCNGeometryPrimitiveType.
primitiveCount: The number of primitives in the element.
bytesPerIndex: The number of bytes that represent a single index value in the data.

Returns a new geometry element object.

An element,Äôs data is an array of index values identifying vertices in a geometry source. SceneKit interprets the data as an array of unsigned integers (whose size is specified by the bytesPerIndex parameter), and then connects the vertices in the order specified by this array, arranged according to the primitiveType parameter. To create a custom SCNGeometryMBS object from the geometry element, use the geometryWithSources() method.

13.18.5 Properties

13.18.6 bytesPerIndex as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The number of bytes that represent each index value in the element,Äôs data.
Notes: An element,Äôs data property holds an array of index values identifying vertices in a geometry source. SceneKit interprets the data as an array of unsigned integers, whose size is specified by the bytesPerIndex property.
(Read only property)

13.18.7 Data as MemoryBlock

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The data describing the geometry element.
Notes: An element,Äôs data is an array of index values identifying vertices in a geometry source. SceneKit interprets the data as an array of unsigned integers, whose size is specified by the bytesPerIndex property.
(Read only property)

13.18.8 Handle as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.18.9 maximumPointScreenSpaceRadius as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The largest radius, measured in screen points, at which to render any point in the geometry element.
Notes: Some visual effects call for rendering a geometry as a collection of individual points—that is, a point cloud, not a solid surface or wireframe mesh. When you use this option, SceneKit can render each point as a small 2D surface that always faces the camera. By applying a texture or custom shader to that surface, you can efficiently render many small objects at once.
To render a geometry element as a point cloud, you must set three properties: pointSize, minimumPointScreenSpaceRadius, and maximumPointScreenSpaceRadius. Use pointSize to determine how large each point appears in world space, so that points farther away appear as smaller 2D surfaces. Use the minimum and maximum radius properties to ensure that the on-screen rendering of each point fits within a certain range of pixel sizes.
For example, to render a point cloud where each point is always one pixel wide (like a field of stars), set both the minimum and maximum sizes to one pixel. To render a group of objects whose screen sizes vary with perspective (like a set of images representing planets), set the minimum size to one pixel and the maximum size to a much larger value.
(Read and Write property)

13.18.10 minimumPointScreenSpaceRadius as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The smallest radius, measured in screen points, at which to render any point in the geometry element.

Notes: Some visual effects call for rendering a geometry as a collection of individual points—that is, a point cloud, not a solid surface or wireframe mesh. When you use this option, SceneKit can render each point as a small 2D surface that always faces the camera. By applying a texture or custom shader to that surface, you can efficiently render many small objects at once.
To render a geometry element as a point cloud, you must set three properties: pointSize, minimumPointScreenSpaceRadius, and maximumPointScreenSpaceRadius. Use pointSize to determine how large each point appears in world space, so that points farther away appear as smaller 2D surfaces. Use the minimum and maximum radius properties to ensure that the on-screen rendering of each point fits within a certain range of pixel sizes.
For example, to render a point cloud where each point is always one pixel wide (like a field of stars), set both the minimum and maximum sizes to one pixel. To render a group of objects whose screen sizes vary with perspective (like a set of images representing planets), set the minimum size to one pixel and the maximum size to a much larger value.
(Read and Write property)

13.18.11 pointSize as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The width of each point in the geometry element, as measured in the geometry's local 3D coordinate space.
Notes: Some visual effects call for rendering a geometry as a collection of individual points—that is, a point cloud, not a solid surface or wireframe mesh. When you use this option, SceneKit can render each point as a small 2D surface that always faces the camera. By applying a texture or custom shader to that surface, you can efficiently render many small objects at once.
To render a geometry element as a point cloud, you must set three properties: pointSize, minimumPointScreenSpaceRadius, and maximumPointScreenSpaceRadius. Use pointSize to determine how large each point appears in world space, so that points farther away appear as smaller 2D surfaces. Use the minimum and maximum radius properties to ensure that the on-screen rendering of each point fits within a certain range of pixel sizes.
(Read and Write property)

13.18.12 primitiveCount as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The number of primitives in the element.
Notes: (Read only property)

13.18.13 primitiveRange as NSRangeMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The range of primitives from the geometry element to render.
Notes: The default value for this property is an NSRangeMBS whose location is NSNotFound (-1) and length is zero, indicating that, by default, SceneKit renders the entire set of primitives specified by a geometry element's data buffer.
You can change a geometry without redefining it by choosing to render only a subset of the primitives specified by a geometry element. To do so, set this property to a subrange of primitive indexes.
(Read and Write property)

13.18.14 primitiveType as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The drawing primitive that connects vertices when rendering the geometry element.
Notes: (Read only property)

13.18.15 Constants

Primitive Types

Constant	Value	Description
PrimitiveTypeLine	2	The geometry element,Äôs data is a sequence of line segments, with each line segment described by two new vertices.
PrimitiveTypePoint	3	The geometry element,Äôs data is a sequence of unconnected points.
PrimitiveTypePolygon	4	The geometry element,Äôs data is a sequence of arbitrary polygons. see https://developer.apple.com/documentation/scenekit/scngeometryprimitivetype/scngeometryprimitivetypepolygon?language=objc
PrimitiveTypeTriangles	0	The geometry element,Äôs data is a sequence of triangles, with each triangle described by three new vertices.
PrimitiveTypeTriangleStrip	1	The geometry element,Äôs data is a sequence of triangles, with each triangle described by one new vertex and two vertices from the previous triangle.

13.19 class SCNGeometryMBS

13.19.1 class SCNGeometryMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A three-dimensional shape (also called a model or mesh) that can be displayed in a scene, with attached materials that define its appearance.
Notes: In SceneKit, geometries attached to SCNNode objects form the visible elements of a scene, and SCNMaterial objects attached to a geometry determine its appearance.

Working with Geometry Objects

You control a geometry,Äôs appearance in a scene with nodes and materials. A geometry object provides only the form of a visible object rendered by SceneKit. You specify color and texture for a geometry,Äôs surface, control how it responds to light, and add special effects by attaching materials (for details, see the methods in Managing a Geometry,Äôs Materials). You position and orient a geometry in a scene by attaching it to an SCNNode object. Multiple nodes can reference the same geometry object, allowing it to appear at different positions in a scene.

You can easily copy geometries and change their materials. A geometry object manages the association between immutable vertex data and a mutable assignment of materials. To make a geometry appear more than once in the same scene with a different set of materials, use its inherited copy method. The copy shares the underlying vertex data of the original, but can be assigned materials independently. You can thus make many copies of a geometry without incurring a significant cost to rendering performance.

You can animate a geometry object. The vertex data associated with a geometry is immutable, but SceneKit provides several ways to animate geometry. You can use a SCNMorpher or SCNSkinner object to deform a geometry,Äôs surface, or run animations created in an external 3D authoring tool and loaded from a scene file. You can also use methods in the SCNShadable protocol to add custom GLSL shader programs that alter SceneKit,Äôs rendering of a geometry.

Obtaining a Geometry Object

SceneKit provides several ways to introduce geometry objects to your app:

Action	For further information
Load from a scene file created using external 3D authoring tools	SCNScene, SCNSceneSource
Use and customize SceneKit,Äôs built-in primitive shapes	SCNPlane, SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, SCNCapsule, SCNTube, and SCNTorus
Create 3D geometry from 2D text or B√©zier curves	SCNText, SCNShape
Create a custom geometry from vertex data	SCNGeometrySource, SCNGeometryElement, geometryWithSources.

Blog Entries

- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 18.2, page 32: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.19.2 Methods

13.19.3 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new geometry object with no content (or default content).
Notes: This method creates a geometry with no visible content. You can use an empty geometry with another geometry,Äôs levelsOfDetail property to make the geometry disappear when it is too far away from the camera to usefully render.
SceneKit,Äôs SCNGeometryMBS subclasses use this method to create geometry instances with default contents. For example, if you call this method on the SCNSphereMBS class, it creates a sphere geometry whose radius property has the default value of 0.5.
You cannot add geometry sources or elements to a geometry object after creating it. To create a custom geometry from your own source and element data, use the geometryWithSources method.

13.19.4 copy as SCNGeometryMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the geometry.

13.19.5 geometry as SCNGeometryMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new geometry object with no content (or default content).
Notes: This method creates a geometry with no visible content. You can use an empty geometry with another geometry,Äôs levelsOfDetail property to make the geometry disappear when it is too far away from the camera to usefully render.
SceneKit,Äôs SCNGeometryMBS subclasses use this method to create geometry instances with default contents. For example, if you call this method on the SCNSphereMBS class, it creates a sphere geometry whose radius property has the default value of 0.5.
You cannot add geometry sources or elements to a geometry object after creating it. To create a custom geometry from your own source and element data, use the geometryWithSources method.

13.19.6 geometryElementAtIndex(index as Integer) as SCNGeometryElementMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the geometry element at a specified index.
Notes: Each SCNGeometryElementMBS object describes how vertices from the geometry,Äôs sources are combined into polygons to create the geometry,Äôs shape. Visible geometries contain at least one element.

13.19.7 geometryElements as SCNGeometryElementMBS()

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of geometry elements that describe the geometry,Äôs shape.
Notes: Each SCNGeometryElementMBS object describes how vertices from the geometry,Äôs sources are combined into polygons to create the geometry,Äôs shape. Visible geometries contain at least one element. For geometries with multiple elements, you can use the materials property to attach different materials to each element.

13.19.8 geometrySources as SCNGeometrySourceMBS()

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of geometry sources that provide vertex data for the geometry.
Notes: Each SCNGeometrySourceMBS object describes an attribute of all vertices in the geometry (such as vertex position, surface normal vector, color, or texture mapping coordinates) identified by the source,Äôs semantic property. A geometry always has at least one source (for the SCNGeometrySourceSemanticVertex semantic), typically has additional sources for use in lighting and shading, and may have other sources for skeletal animation or surface subdivision information.

13.19.9 geometrySourcesForSemantic(semantic as String) as SCNGeometrySourceMBS()

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the geometry sources for a specified semantic.
Notes: semantic: A constant identifying a semantic for which to return geometry sources. See Geometry Semantic Identifiers for possible values.

Returns an array of SCNGeometrySource objects, or nil if the geometry has no source for the specified semantic.

Each SCNGeometrySourceMBS object describes an attribute of all vertices in the geometry (such as vertex position, surface normal vector, color, or texture mapping coordinates) identified by the source,Äôs semantic

property. A geometry always has at least one source, for the SCNGeometrySourceSemanticVertex semantic, typically has additional sources for use in lighting and shading, and may have other sources for skeletal animation or surface subdivision information.
The vertex, normal, and color semantics each refer to at most one source. A geometry may have multiple sources for the SCNGeometrySourceSemanticTexcoord semantic—in this case, indices in the returned array correspond to values for the mappingChannel property used when attaching textures to materials.

13.19.10 geometryWithSources(sources() as SCNGeometrySourceMBS, elements() as SCNGeometryElementMBS) as SCNGeometryMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new geometry built from the specified geometry sources and elements.
Notes: sources: An array of SCNGeometrySourceMBS objects describing vertices in the geometry and their attributes.
elements: An array of SCNGeometryElementMBS objects describing how to connect the geometry,Äôs vertices.

A geometry,Äôs visible content comes from the combination of geometry sources, which contain data describing its vertices, with geometry elements, which contain data describing how the vertices connect to form a surface.
Each SCNGeometrySourceMBS object describes an attribute of all vertices in the geometry (vertex position, surface normal vector, color, or texture mapping coordinates) identified by the source,Äôs semantic property. To create a custom geometry you must provide at least one source, for the SCNGeometrySourceSemanticVertex semantic. Typically, you also provide sources for normals and texture coordinates for use in lighting and shading.
Sources for the vertex, normal, and color semantics must be unique—if multiple objects in the sources array have the same semantic, SceneKit uses only the first. A geometry may have multiple sources for the SCNGeometrySourceSemanticTexcoord semantic—the order of texture coordinate sources in the sources array determines the value to use for the mappingChannel property when attaching materials.
Each SCNGeometryElementMBS object describes how vertices from the geometry sources are combined into polygons to create the geometry,Äôs shape. Creating a custom geometry requires at least one element. If the elements array contains multiple objects, their order determines the arrangement of the geometry,Äôs materials—for details, see the discussion of the materials property.

13.19.11 insertMaterial(Material as SCNMaterialMBS, Index as Integer)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Insert a material in the materials array at the specified index.
Notes: material: The material to insert.
index: Index in the materials array to insert the new material.

13.19.12 levelsOfDetail as SCNLevelOfDetailMBS()

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: An array of SCNLevelOfDetail objects for managing the geometry,Äôs appearance when viewed from far away.
Notes: Because rendering a complex geometry incurs a performance cost, you can use level-of-detail objects to substitute simpler geometries in its place as its distance from the point of view camera increases (or its apparent size decreases). For details, see SCNLevelOfDetailMBS.

13.19.13 materials as SCNMaterialMBS()

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the receiver's materials array.
Notes: Each geometry element can be rendered using a different material. The index of the material used for a geometry element is equal to the index of that element modulo the number of materials.

13.19.14 materialWithName(name as string) as SCNMaterialMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Return the first material from the materials array of the receiver with the specified name.
Notes: name: The name of the material to retrieve.

13.19.15 removeMaterial(Index as Integer)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Remove the material at the specified index from the materials array.
Notes: index: The index of the material to remove from the 'materials' array.

13.19.16 replaceMaterial(Index as Integer, Material as SCNMaterialMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Remove the material at the index 'index' from the materials array of the receiver and insert 'material' in its position.
Notes: index: The index of the material to replace in the materials array.
material: The new material that will replace the previous one.

13.19.17 setLevelsOfDetail(LevelsOfDetails() as SCNLevelOfDetailMBS = nil)

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Sets an array of SCNLevelOfDetail objects for managing the geometry,Äôs appearance when viewed from far away.
Notes: Value can be nil to clear it.

13.19.18 setMaterials(materials() as SCNMaterialMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the materials.

13.19.19 Properties

13.19.20 edgeCreasesElement as SCNGeometryElementMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The geometry element identifying which edges of the geometry,Äôs surface should remain sharp after subdivision.
Notes: This geometry element,Äôs primitiveType value must be SCNGeometryPrimitiveTypeLine. The geometry element,Äôs data is an array of vertex indices, each pair of which defines a line segment identifying an edge to be treated as a crease during subdivision. Use the edgeCreasesSource property to specify the smoothness or sharpness of each crease.
(Read and Write property)

13.19.21 edgeCreasesSource as SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The geometry source specifying the smoothness or sharpness of edges after surface subdivision.
Notes: This geometry source,Äôs semantic value must be SCNGeometrySourceSemanticEdgeCrease. Its data is an array of scalar values (that is, the source,Äôs componentsPerVector value is 1). The value at an index in the geometry source determines the smoothness or sharpness of the edge identified by the primitive at the corresponding index in the edgeCreasesElement geometry element: a value of 0.0 specifies a completely smoothed edge, and a value of 10.0 or greater specifies an infinitely sharp edge.
(Read and Write property)

13.19.22 firstMaterial as SCNMaterialMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines the first material of the geometry. Returns nil if the geometry has no material.
Notes: This method is here for convenience. It is equivalent to the first object in the "materials" array above.
(Read and Write property)

13.19.23 geometryElementCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the number of geometry elements owned by the geometry.
Notes: (Read only property)

13.19.24 Handle as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.19.25 Name as String

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A name associated with the geometry object.
Notes: You can provide a descriptive name for a geometry object to make managing your scene graph easier. Geometries loaded from a scene file may have names assigned by an artist using a 3D authoring tool. Use the SCNSceneSourceMBS class to examine geometries in a scene file without loading its scene graph.

Geometry names are saved when you export a scene to a file using its writeToURL method. They also appear in the Xcode scene editor.
(Read and Write property)

13.19.26 subdivisionLevel as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the subdivision level of the receiver.
Notes: Defaults to 0.
A subdivision level of 0 means no subdivision. When the ‘tessellator‘ property of the receiver is not nil, the refinement is done on the GPU.
(Read and Write property)

13.19.27 tessellator as SCNGeometryTessellatorMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies how the geometry should be tessellated at render time on the GPU.
Notes: Defaults to nil.
(Read and Write property)

13.19.28 wantsAdaptiveSubdivision as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies if the subdivision is adaptive or uniform.
Notes: Defaults to true.
Adaptive subdivision requires that the ‘tessellator‘ property of the receiver is not nil.
(Read and Write property)

13.20 class SCNGeometrySourceMBS

13.20.1 class SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: A container for vertex data forming part of the definition for a three-dimensional object, or geometry.
Notes: You use geometry sources together with SCNGeometryElement objects to define custom SCNGeometry objects or to inspect the data that composes an existing geometry.

see
https://developer.apple.com/documentation/scenekit/scngeometrysource
Blog Entries

- MBS Xojo Plugins, version 22.1pr1
- MBS Xojo Plugins, version 19.3pr6
- MBS Xojo Plugins, version 19.3pr5

13.20.2 Methods

13.20.3 Constructor

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

13.20.4 geometrySourceWithData(data as MemoryBlock, Semantic as String, vectorCount as Integer, floatComponents as boolean, componentsPerVector as Integer, bytesPerComponent as Integer, dataOffset as Integer, dataStride as Integer) as SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry source from the specified data and options.
Notes: data: The data for the geometry source.
semantic: The semantic value (or attribute) that the geometry source describes for each vertex. See Geometry Semantic Identifiers for available values.
vectorCount: The number of geometry source vectors.
floatComponents: A Boolean value that indicates whether vector components are floating-point values. Specify true for floating-point values, or false for integer values.
componentsPerVector: The number of scalar components in each vector.

bytesPerComponent: The size, in bytes, of each vector component.
offset: The offset, in bytes, from the beginning of the data to the first vector component to be used in the geometry source.
stride: The number of bytes from each vector to the next in the data.

Returns a new geometry source object.

sA geometry source,Äôs data is an array of vectors, each of which represents a particular attribute (or semantic) of a vertex in the geometry. The other parameters determine how SceneKit interprets this data. For example, an array of vertex positions may have three 32-bit floating-point components per vector, but an array of texture coordinates may have two 8-bit integer coponents per vector. You can use the offset and stride parameters together to interleave data for multiple geometry sources in the same array, improving rendering performance. See SCNGeometrySourceMBS for details.
To create a custom SCNGeometryMBS object from the geometry source, use the geometryWithSources method.

13.20.5 geometrySourceWithNormals(Normals() as SCNVector3MBS) as SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry source from an array of normal vectors.
Notes: normals: An array of three-component vectors, each of which represents a surface normal for the geometry source.

A new geometry source whose SCNGeometrySource property is SCNGeometrySourceSemanticNormal.

SceneKit converts this data to its own format to optimize rendering performance. To read the converted data, examine the properties of the created SCNGeometrySourceMBS object.
To create a custom SCNGeometryMBS object from the geometry source, use the geometryWithSources method.
See also:

- 13.20.6 geometrySourceWithNormals(texcoord() as CGPointMBS) as SCNGeometrySourceMBS 950

13.20.6 geometrySourceWithNormals(texcoord() as CGPointMBS) as SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry source from an array of texture coordinate points.
Notes: texcoord: An array of points, each of which represents a texture coordinate pair for the geometry

source.

A new geometry source whose SCNGeometrySource property is SCNGeometrySourceSemanticTexcoord.

SceneKit converts this data to its own format to optimize rendering performance. To read the converted data, examine the properties of the created SCNGeometrySourceMBS object.
To create a custom SCNGeometryMBS object from the geometry source, use the geometryWithSources:elements: method.
See also:

- 13.20.5 geometrySourceWithNormals(Normals() as SCNVector3MBS) as SCNGeometrySourceMBS 950

13.20.7 geometrySourceWithVertices(vertices() as SCNVector3MBS) as SCNGeometrySourceMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a geometry source from an array of vertex positions.
Notes: vertices: An array of three-component vectors, each of which represents a vertex position for the geometry source.

Returns a new geometry source whose SCNGeometrySource property is SCNGeometrySourceSemanticVertex.

SceneKit converts this data to its own format to optimize rendering performance. To read the converted data, examine the properties of the created SCNGeometrySourceMBS object.
To create a custom SCNGeometryMBS object from the geometry source, use the geometryWithSources method.

13.20.8 SCNGeometrySourceSemanticBoneIndices as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for bone index data, used for skeletal animation of skinned surfaces.
Notes: For a geometry source, this semantic identifies data containing skeletal animation data for each vertex in the geometry. SceneKit uses this information to determine which bone nodes in the skeleton affect the behavior of each vertex.
For a custom shader program, you use this semantic to bind SceneKit,Äôs bone index data to an input attribute of the shader.
For details on skeletal animation, see SCNSkinnerMBS.

13.20.9 SCNGeometrySourceSemanticBoneWeights as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for bone weight data, used for skeletal animation of skinned surfaces.
Notes: For a geometry source, this semantic identifies data containing skeletal animation data for each vertex in the geometry. SceneKit uses this information to determine how much a vertex,Äôs position is influenced by the positions of bone nodes in the skeleton.
For a custom shader program, you use this semantic to bind SceneKit,Äôs bone weight data to an input attribute of the shader.
For details on skeletal animation, see SCNSkinnerMBS.

13.20.10 SCNGeometrySourceSemanticColor as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for per-vertex color data.
Notes: For a geometry source, this semantic identifies data containing a color for each vertex in the geometry. SceneKit interpolates per-vertex colors across a surface to produce smooth shading. Per-vertex colors modulate those produced by lighting and a geometry,Äôs materials, if applicable.
For a custom shader program, you use this semantic to bind SceneKit,Äôs vertex color data to an input attribute of the shader.
Vertex color data is typically an array of three- or four-component vectors.

13.20.11 SCNGeometrySourceSemanticEdgeCrease as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for edge crease data, used for subdividing surfaces.
Notes: For a geometry source, this semantic identifies data containing crease data for each vertex in the geometry. SceneKit uses this information to determine the sharpness of edges and smoothness of surfaces when you change a geometry,Äôs subdivisionLevel property.
For a custom shader program, you use this semantic to bind SceneKit,Äôs edge crease data to an input attribute of the shader.
Edge crease data is an array of scalar floating-point values, where each value determines the smoothness or sharpness of the edge identified by the primitive at the corresponding index in the geometry,Äôs SceneKit Constants geometry element: A value of 0.0 specifies a completely smoothed edge, and a value of 10.0 or greater specifies an infinitely sharp edge.

13.20.12 SCNGeometrySourceSemanticNormal as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for surface normal data.
Notes: For a geometry source, this semantic identifies data containing the surface normal vector at each vertex in the geometry. SceneKit uses this information to compute lighting effects on the surface.
For a custom shader program, you use this semantic to bind SceneKit,Äôs vertex normal data to an input attribute of the shader.
Vertex normal data is typically an array of three- or four-component vectors.

13.20.13 SCNGeometrySourceSemanticTangent as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for surface tangent vector data.
Notes: For a geometry source, this semantic identifies data containing the surface tangent vector at each vertex in the geometry. SceneKit uses this information to compute advanced lighting effects on the surface.
For a custom shader program, you use this semantic to bind SceneKit,Äôs vertex tangent data to an input attribute of the shader.
Vertex tangent data is typically an array of three- or four-component vectors.

13.20.14 SCNGeometrySourceSemanticTexcoord as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for texture coordinate data.
Notes: For a geometry source, this semantic identifies data containing texture mapping coordinates for each vertex in the geometry. Unlike other semantics, a geometry may contain multiple sources for texture coordinates—each corresponds to a separate mappingChannel number that you can use when associating textured materials.
For a custom shader program, you use this semantic to bind SceneKit,Äôs texture coordinate data to one or more input attributes of the shader.
Texture coordinate data is typically an array of two-component vectors.

13.20.15 SCNGeometrySourceSemanticVertex as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for vertex position data.
Notes: For a geometry source, this semantic identifies data containing the positions of each vertex in the geometry. If you create a custom geometry using the geometryWithSources method, you must provide a geometry source for this semantic.
For a custom shader program, you use this semantic to bind SceneKit,Äôs vertex position data to an input attribute of the shader.
Vertex position data is typically an array of three- or four-component vectors.

13.20.16 SCNGeometrySourceSemanticVertexCrease as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic for vertex crease data, used for subdividing surfaces.
Notes: For a geometry source, this semantic identifies data containing crease data for each vertex in the geometry. SceneKit uses this information to determine the sharpness of corners and smoothness of surfaces when you change a geometry,Äôs subdivisionLevel property.
For a custom shader program, you use this semantic to bind SceneKit,Äôs vertex crease data to an input attribute of the shader.
Vertex crease data is an array of scalar floating-point values, where each value determines the smoothness or sharpness of the corresponding vertex: A value of 0.0 specifies a completely smoothed corner, and a value of 10.0 or greater specifies an infinitely sharp point.

13.20.17 Properties

13.20.18 bytesPerComponent as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The size, in bytes, of each vector component.
Notes: (Read only property)

13.20.19 componentsPerVector as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The number of scalar components in each vector.
Notes: (Read only property)

13.20.20 Data as MemoryBlock

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The data for the geometry source.
Notes: A geometry source,Äôs data is an array of vectors, each of which represents a particular attribute (or semantic) of a vertex in the geometry. The other properties of the geometry source determine how SceneKit interprets this data. For example, an array of vertex positions may have three 32-bit floating-point components per vector, but an array of texture coordinates may have two 8-bit integer coponents per vector. (Read only property)

13.20.21 dataOffset as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The offset, in bytes, from the beginning of the data to the first vector component to be used in the geometry source.
Notes: You can use the dataOffset and dataStride parameters can be used together to interleave data for multiple geometry sources in the same array, improving rendering performance. See SCNGeometrySourceMBS for details.
(Read only property)

13.20.22 dataStride as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The number of bytes from a vector to the next one in the data.
Notes: You can use the dataOffset and dataStride parameters can be used together to interleave data for multiple geometry sources in the same array, improving rendering performance. See SCNGeometrySourceMBS for details.
(Read only property)

13.20.23 floatComponents as Boolean

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether vector components are floating-point values.
Notes: If true, SceneKit interprets the geometry source,Äôs data as an array of vectors whose components are floating-point values. The type of floating-point value is determined by the SCNGeometrySourceMBS property: 4 bytes for float values or 8 bytes for double values.
If false, SceneKit interprets the geometry source,Äôs data as an array of vectors whose components are integer values. The type of integer value is determined by the SCNGeometrySourceMBS property; for example, 2 bytes for unsigned short values or 4 bytes for unsigned int values.
(Read only property)

13.20.24 Handle as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.20.25 semantic as String

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The semantic value (or attribute) the geometry source describes for each vertex.
Notes: A semantic describes an attribute for each vertex, such as position, color, surface normal vector, or texture coordinates.
See Geometry Semantic Identifiers for available values.
(Read only property)

13.20.26 vectorCount as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The number of vectors in the data.
Notes: (Read only property)

13.21 class SCNGeometryTessellatorMBS

13.21.1 class SCNGeometryTessellatorMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: A geometry tessellator describes how a more detailed surface is calculated from the geometry's initial surface.
Blog Entries

- MBS Xojo Plugins, version 19.3pr5

13.21.2 Methods

13.21.3 Constructor

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

13.21.4 copy as SCNGeometryTessellatorMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

13.21.5 Properties

13.21.6 Adaptive as Boolean

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies if the tessellation should be uniform or adaptive.
Notes: Defaults to false.
(Read and Write property)

13.21.7 edgeTessellationFactor as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the edge tessellation factor.
Notes: Defaults to 1.
This has no effect for adaptive subdivision.
(Read and Write property)

13.21.8 Handle as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.21.9 insideTessellationFactor as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the inside tessellation factor.
Notes: Defaults to 1.
This has no effect for adaptive subdivision
(Read and Write property)

13.21.10 maximumEdgeLength as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the maximum edge length.
Notes: Defaults to 1.
This has no effect for non-adaptive subdivision
(Read and Write property)

13.21.11 ScreenSpace as Boolean

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies if the level of tessellation should be adapted in screenSpace.
Notes: Defaults to false.
(Read and Write property)

13.21.12 smoothingMode as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The smoothing mode.
Notes: Defaults to SmoothingModeNone.
(Read and Write property)

13.21.13 tessellationFactorScale as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the scale factor applied to the per-patch tessellation factors.
Notes: Defaults to 1.
(Read and Write property)

13.21.14 tessellationPartitionMode as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the tessellation partition mode.
Notes: Defaults to TessellationPartitionModeInteger.

see also
https://developer.apple.com/documentation/metal/mtltessellationpartitionmode?language=objc
(Read and Write property)

13.21.15 Constants

Smoothing Modes

Constant	Value	Description
SmoothingModeNone	0	
SmoothingModePhong	2	
SmoothingModePNTriangles	1	

Tessellation Partition Modes

Constant	Value	Description
TessellationPartitionModeFractionalEven	3	A fractional even partitioning mode.
TessellationPartitionModeFractionalOdd	2	A fractional odd partitioning mode.
TessellationPartitionModeInteger	1	An integer partitioning mode.
TessellationPartitionModePow2	0	A power of two partitioning mode.

13.22 class SCNHitTestResultMBS

13.22.1 class SCNHitTestResultMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Information about the result of a scene-space or view-space search for scene elements.
Notes: Hit-testing is the process of finding elements of a scene located at a specified point, or along a specified line segment (or ray). An SCNHitTestResult object provides details about one result from a hit-test search. There are three ways to perform a hit-test search. Use the hitTest method of an SCNViewMBS object (or other scene renderer), the hitTestWithSegment method of a node, or the rayTestWithSegment method of your scene,Äôs physics world.

When you perform a hit-test search, SceneKit looks for SCNGeometryMBS objects along the ray you specify. For each intersection between the ray and and a geometry, SceneKit creates a hit-test result to provide information about both the SCNNodeMBS object containing the geometry and the location of the intersection on the geometry,Äôs surface.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.2
- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 20.2pr1

Videos

- XDC 2020 MBS Plugins Presentation

13.22.2 Methods

13.22.3 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.22.4 SCNHitTestBackFaceCullingKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to ignore faces not oriented toward the camera.
Notes: The value for this key is a Boolean value. The default value is true, specifying that back-facing polygons will not be returned as hit-test results.

13.22.5 SCNHitTestBoundingBoxOnlyKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to search for objects by bounding box only.
Notes: The value for this key is a Boolean value. The default value is false, specifying that hit-testing searches should test against node geometry. Specifying true for this option increases search performance at the expense of geometric accuracy.

13.22.6 SCNHitTestClipToZRangeKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to search for objects only within the depth range of the current point of view.
Notes: The value for this key is a Boolean value. The default value is true, specifying that hit-testing searches only objects between the zNear and zFar distances of the pointOfView camera. Specify false to include objects outside this depth range in the search.
This option is valid only when hit-testing in the screen space of an SCNSceneRenderer object with the hitTest method.

13.22.7 SCNHitTestIgnoreChildNodesKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to ignore child nodes when searching.
Notes: The value for this key is a Boolean value. The default value is false, specifying that hit-testing may return objects from any portion of the node hierarchy. Specify true to search only the node specified by the SCNHitTestRootNodeKey key.

13.22.8 SCNHitTestIgnoreHiddenNodesKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to ignore hidden nodes when searching.
Notes: The value for this key is a Boolean value. The default value is true, specifying that hit-testing should not return nodes whose hidden property value is true. Specify false to search nodes regardless of their visibility.

13.22.9 SCNHitTestOptionCategoryBitMask as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option to search only for objects matching a specified bitmask.
Notes: The value for this key is an integer value. If present, the search will return only nodes that both satisfy the hit test and have a categoryBitMask value overlapping this bitmask.

13.22.10 SCNHitTestOptionSearchMode as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option for the number and order of hit test results to provide.
Notes: The value for this key is an raw integer value of an SCNHitTestSearchMode constant.

13.22.11 SCNHitTestRootNodeKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The root of the node hierarchy to be searched.
Notes: The value for this key is an SCNNodeMBS object. Hit-testing searches only the child node hierarchy under this node. When hit-testing takes place in the screen space of an SCNSceneRenderer object with the hitTest() method, the default value is the presented scene,Äôs root node. When hit-testing is in a node using its hitTestWithSegment(), the default value is the node.

13.22.12 Properties

13.22.13 boneNode as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The hit bone.
Notes: Only available if the node hit has a SCNSkinnerMBS attached.
(Read only property)

13.22.14 faceIndex as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The index of the primitive in the geometry element intersected by the search ray.
Notes: (Read only property)

13.22.15 geometryIndex as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The index of the geometry element whose surface the search ray intersects.
Notes: Every SCNGeometryMBS object contains one or more SCNGeometryElementMBS objects that define how its vertices connect to form a surface. This property provides the index of the geometry element intersecting the search ray. For more information about that geometry element, use the geometry,Äôs geometryElementAtIndex method.
(Read only property)

13.22.16 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.22.17 localCoordinates as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point of intersection between the geometry and the search ray, in the local coordinate system of the node containing the geometry.
Notes: (Read only property)

13.22.18 localNormal as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The surface normal vector at the point of intersection, in the local coordinate system of the node containing the geometry intersected by the search ray.
Notes: (Read only property)

13.22.19 modelTransform as SCNMatrix4MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The world transform matrix of the node containing the intersection.
Notes: Use this matrix to transform vectors from the local coordinate space of the node whose geometry is intersected by the search ray to the scene,Äôs world coordinate system.

(Read only property)

13.22.20 node as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The node whose geometry intersects the search ray.
Notes: (Read only property)

13.22.21 worldCoordinates as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point of intersection between the geometry and the search ray, in the scene,Äôs world coordinate system.
Notes: (Read only property)

13.22.22 worldNormal as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The surface normal vector at the point of intersection, in the scene,Äôs world coordinate system.
Notes: (Read only property)

13.22.23 Constants

Hit Test Search Modes

Constant	Value	Description
SCNHitTestSearchModeAll	1	The hit test should return all possible results, sorted from nearest to farthest.
SCNHitTestSearchModeAny	2	The hit test should return only the first object found, regardless of distance.
SCNHitTestSearchModeClosest	0	The hit test should return only the closes object found.

13.23 class SCNIKConstraintMBS

13.23.1 class SCNIKConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A constraint that applies inverse kinematics to make a chain of nodes ‚Äúreach‚Äù toward a target point.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnikconstraint
Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.23.2 Methods

13.23.3 Constructor(chainRootNode as SCNNodeMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNIKConstraint object with the specified parameter.
Notes: chainRootNode The root node of the kinematic chain.
"chainRootNode" must be an ancestor of the node on which the constraint is applied.

13.23.4 inverseKinematicsConstraintWithChainRootNode(chainRootNode as SCNNodeMBS) as SCNIKConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNIKConstraint object with the specified parameter.
Notes: chainRootNode The root node of the kinematic chain.
"chainRootNode" must be an ancestor of the node on which the constraint is applied.

13.23.5 Properties

13.23.6 chainRootNode as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the root node of the kinematic chain.
Notes: (Read only property)

13.23.7 targetPosition as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the target position (in world space coordinates) of the end joint (i.e the node that owns the IK constraint).
Notes: Defaults to (0,0,0). Animatable.
(Read and Write property)

13.24 control SCNIOSControlMBS

13.24.1 control SCNIOSControlMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The Xojo control for a SCNViewMBS.
Notes: This control embeds a special SCNView subclass.
Designed for Xojo 2020r2 and newer.
Please use view property to access the underlaying object and set properties.

To learn about rendering events, please check here:
https://developer.apple.com/documentation/scenekit/scnscenerendererdelegate?language=objc

You can set PreferLowPowerDevice and PreferredRenderingAPI properties before initializing a view.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5
- MBS Xojo Plugins, version 23.5pr7
- MBS Xojo Plugins Version 21.0 News
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0
- Three new controls for iOS in Xojo
- MBS Xojo Plugins, version 20.6pr3

Xojo Developer Magazine

- 22.1, page 9: News
- 19.2, page 9: News

13.24.2 Properties

13.24.3 View as SCNViewMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The view used in the control.
Notes: Use this object to set more options on the control.
(Read only property)

13.24.4 Events

13.24.5 Close

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control is about to close.

13.24.6 GotFocus

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control itself got focus.
Notes: This only fires if the control itself got focus and not a sub control.

13.24.7 LostFocus

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control lost focus.
Notes: This only fires if the control itself lost focus and not a sub control.

13.24.8 Open

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control is about to was created and you can initialize it.

13.24.9 rendererDidApplyAnimations(time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Tells the control to perform any updates that need to occur after actions and animations are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this method to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.24.10 rendererDidApplyConstraints(time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked on the control once the scene renderer did apply the constraints.
Notes: time The time at which the constraints were simulated.

All modifications done within this method don't go through the transaction model, they are directly applied on the presentation tree.

13.24.11 rendererDidRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Tells the control that the renderer has cleared the viewport and is about to render the scene.
Notes: scene: The SCNSceneMBS object to be rendered.
time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this event to perform custom drawing before SceneKit renders a scene—for example, to draw backdrop content underneath SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this event—the results of modifying SceneKit objects during this event are undefined.

To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this event. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.24.12 rendererDidSimulatePhysics(time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Tells the control to perform any updates that need to occur after physics simulations are performed.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this event exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

This event is the last opportunity SceneKit provides for you to change the scene graph before rendering.

13.24.13 rendererUpdate(time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Tells the control to perform any updates that need to occur before actions, animations, and physics are evaluated.
Notes: time: The current system time, in seconds. Use this parameter for any time-based elements of your game logic.

SceneKit calls this method exactly once per frame, so long as the SCNViewMBS object (or other SCNSceneRenderer object) displaying the scene is not paused.

Implement this event to add game logic to the rendering loop. Any changes you make to the scene graph during this method are immediately reflected in the displayed scene. That is, SceneKit immediately updates the hierarchy of presentation nodes it uses to render the scene (instead of using the SCNTransactionMBS class to ,Äúbatch,Äù your changes).

13.24.14 rendererWillRenderScene(scene as SCNSceneMBS, time as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Tells the control that the renderer has rendered the scene.
Notes: scene: The scene object that was rendered.

time: The current system time, in seconds. If your custom rendering involves animation, use this parameter to compute your own animation state.

Implement this method to perform custom drawing after SceneKit has rendered a scene—for example, to draw overlay content on top of SceneKit content. You should only execute Metal or OpenGL drawing commands (and any setup required to perform them) in this method—the results of modifying SceneKit objects in this method are undefined.

- To render using Metal, use the renderer parameter to retrieve the scene renderer,Äôs currentRenderCommandEncoder object and encode your own drawing commands. If you need to reference other Metal state, see the properties listed in SCNSceneRenderer.
- To render using OpenGL, simply call the relevant OpenGL drawing commands—SceneKit automatically makes its OpenGL context the current context before calling this method. If you need to reference the OpenGL context being rendered into, examine the context property of the renderer parameter.

You must draw using the appropriate graphics technology for the view currently being rendered. Use the renderingAPI property of the renderer object to determine whether Metal or OpenGL is in use.

13.24.15 touchesBegan(e as NSEventMBS, touches() as NSTouchMBS) as boolean

Plugin Version: 23.5, Platform: iOS, Targets: .

Function: Tells this object that one or more new touches occurred in a view or window.
Notes: e: The event to which the touches belong.

UIKit calls this method when a new touch is detected in a view or window. Many UIKit classes override this method and use it to handle the corresponding touch events. The default implementation of this method forwards the message up the responder chain.

Return true if you handled it as otherwise we pass it on.

13.24.16 touchesCancelled(e as NSEventMBS, touches() as NSTouchMBS) as boolean

Plugin Version: 23.5, Platform: iOS, Targets: .

Function: Tells the responder when a system event (such as a system alert) cancels a touch sequence.
Notes: e: The event to which the touches belong.

UIKit calls this method when it receives a system interruption requiring cancellation of the touch sequence. An interruption is anything that causes the application to become inactive or causes the view handling the touch events to be removed from its window. Your implementation of this method should clean up any state associated with handling the touch sequence.

Return true if you handled it as otherwise we pass it on.

13.24.17 touchesEnded(e as NSEventMBS, touches() as NSTouchMBS) as boolean

Plugin Version: 23.5, Platform: iOS, Targets: .

Function: Tells the responder when one or more fingers are raised from a view or window.
Notes: e: The event to which the touches belong.

UIKit calls this method when a finger or Apple Pencil is no longer touching the screen. Many UIKit classes override this method and use it to clean up state involved in the handling of the corresponding touch events. The default implementation of this method forwards the message up the responder chain.

Return true if you handled it as otherwise we pass it on.

13.24.18 touchesMoved(e as NSEventMBS, touches() as NSTouchMBS) as boolean

Plugin Version: 23.5, Platform: iOS, Targets: .

Function: Tells the responder when one or more touches associated with an event changed.
Notes: e: The event to which the touches belong.

UIKit calls this method when the location or force of a touch changes. Many UIKit classes override this method and use it to handle the corresponding touch events. The default implementation of this method forwards the message up the responder chain.

Return true if you handled it as otherwise we pass it on.

13.25 class SCNLevelOfDetailMBS

13.25.1 class SCNLevelOfDetailMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: An alternate resolution for a geometry that SceneKit automatically substitutes to improve rendering performance.
Notes: You use level-of-detail objects when you have a detailed geometry that appears at several apparent sizes in a scene. For example, the teapot model on the the left in the figure below has 256 polygons, the model at center has 1024 polygons, and the model on the right has 14,400 polygons. If all three models appear close to the camera, filling most of the rendered view, the difference in detail between them is clearly visible—but if they appear far away, taking up a small area of the view, the difference is much less obvious. Rendering higher-resolution geometries incurs a higher performance cost.

When you associate one or more level-of-detail objects with a SCNGeometryMBS object using its levelsOfDetail property, SceneKit automatically substitutes alternate geometries when appropriate. For example, the two lower-resolution teapot models seen above can be added as levels of detail for the high-resolution model. For each level of detail, you specify either a world-space distance or a screen-space radius. The measure you specify determines the threshold where SceneKit automatically renders that level of detail,Äôs alternate geometry instead of the original geometry. If you specify a distance, the alternate geometry appears when the node containing the geometry is moved that distance away from the camera. If you specify a radius, the alternate geometry appears when the pixel area covered by the rendered by the geometry is smaller than a circle of that radius.
The geometries associated with lower levels of detail need not share all attributes of the original geometry. For example, you can use different materials for levels of detail that only appear when far away from the camera, disabling expensive features such as per-pixel lighting, reflection mapping, or custom shader programs.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.3pr5

13.25.2 Methods

13.25.3 Constructor

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.25.4 copy as SCNLevelOfDetailMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the object.

13.25.5 levelOfDetailWithGeometryAndScreenSpaceRadius(geo as SCNGeometryMBS, screenSpaceRadius as Double) as SCNLevelOfDetailMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a level of detail with the specified geometry and threshold pixel radius.
Notes: geometry: The geometry to render for this level of detail.
radius: The maximum radius (in pixels) of the geometry,Äôs bounding sphere for this level of detail to appear.

Returns a level-of-detail object. You associate levels of detail with a SCNGeometry object using its levelsOfDetail property.

When rendering a geometry with associated levels of detail, SceneKit calculates the radius in pixels of the circle covered by a geometry,Äôs bounding sphere, then renders the geometry for the SCNLevelOfDetail object with the largest radius parameter smaller than that circle.
If you pass nil for the geometry parameter, SceneKit renders no geometry for the level of detail. Creating a level-of-detail object with no geometry allows you to skip rendering costs entirely for an object when it would appear very far away or very small.

13.25.6 levelOfDetailWithGeometryAndWorldSpaceDistance(geo as SCNGeometryMBS, worldSpaceDistance as Double) as SCNLevelOfDetailMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a level of detail with the specified geometry and threshold camera distance.
Notes: geometry: The geometry to render for this level of detail, or nil if SceneKit should render no geometry at this level of detail.
distance: The minimum distance from the current point of view for this level of detail to appear.

Returns a level-of-detail object. You associate levels of detail with a SCNGeometry object using its levelsOfDetail property.

When rendering a geometry with associated levels of detail, SceneKit calculates the distance from the current point of view to the geometry,Äôs parent node, then renders the geometry for the SCNLevelOfDetail object with the smallest distance parameter greater than that distance.
If you pass nil for the geometry parameter, SceneKit renders no geometry for the level of detail. Creating a level-of-detail object with no geometry allows you to skip rendering costs entirely for an object when it would appear very far away or very small.

13.25.7 Properties

13.25.8 geometry as SCNGeometryMBS

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The geometry associated with this level of detail.
Notes: SceneKit renders this geometry instead of the original geometry when the level of detail is appropriate. Generally, levels of detail with larger worldSpaceDistance values or smaller screenSpaceRadius values should contain less complex geometries.
If the value of this property is nil, SceneKit renders no geometry at this level of detail.
(Read only property)

13.25.9 Handle as Integer

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object handle.
Notes: (Read and Write property)

13.25.10 screenSpaceRadius as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum radius (in pixels) of the geometry,Äôs bounding sphere for this level of detail to appear.
Notes: When rendering a geometry with associated levels of detail, SceneKit calculates the radius in pixels of the circle covered by a geometry,Äôs bounding sphere, then renders the geometry for the SCNLevelOfDetail object with the smallest radius parameter larger than that circle.
(Read only property)

13.25.11 worldSpaceDistance as Double

Plugin Version: 19.3, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum distance from the current point of view for this level of detail to appear.
Notes: When rendering a geometry with associated levels of detail, SceneKit calculates the distance from the current point of view to the geometry,Äôs parent node, then renders the geometry for the SCNLevelOfDetailMBS object with the largest distance parameter less than that distance.
(Read only property)

13.26 class SCNLightMBS

13.26.1 class SCNLightMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A light source that can be attached to a node to illuminate the scene.
Notes: You illuminate your scene by attaching lights to SCNNode objects using their light property.

You set a light,Äôs type using its type property. Depending on a light,Äôs type, its position and direction may affect its behavior—you control the light,Äôs position and direction through the node that the light is attached to. The direction of a light, if applicable to its type, is along the negative z-axis of its node,Äôs local coordinate system.

A light,Äôs other properties affect how it illuminates a scene. All lights have a color property, which interacts with SCNMaterial objects to produce the pixel colors in a rendered scene. Other properties, such as attenuation, shadowing, and spot angle, can affect the behavior of certain types of lights.

The number and type of lights in a scene is a key factor in SceneKit,Äôs rendering performance. For efficient rendering, follow these tips:

- Use SceneKit lights only for dynamic light sources or lights that affect moving objects. For statically lit portions of your scene, create a light map texture in an external 3D authoring tool (also known as baked lighting) and apply it to objects in the scene using the multiply material property.
- Minimize the number of lights on each element of the scene. You can achieve most common lighting effects using no more than three lights, and you only need a single ambient light source. SceneKit only uses up to eight light sources per node when rendering, ignoring any additional lights. If you set the attenuationEndDistance property on a spotlight or omnidirectional light to limit its area of effect, SceneKit ignores the light (and its performance cost) when rendering objects outside that area. You can also use the categoryBitMask property to choose which nodes are illuminated by a light.

Requires MacOS 10.8 or newer.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 18.5pr3

Xojo Developer Magazine

- 19.6, pages 48 to 50: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes

13.26.2 Methods

13.26.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new light object.

13.26.4 copy as SCNLightMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the light.

13.26.5 light as SCNLightMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new light object.

13.26.6 SCNLightTypeAmbient as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A light that illuminates all objects in the scene from all directions.
Notes: Because the intensity of light from an ambient source is the same everywhere in the scene, its position and direction have no effect. Attenuation, spotlight angle, and shadow attributes do not apply to ambient lights.

13.26.7 SCNLightTypeDirectional as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A light source with a uniform direction and constant intensity.
Notes: Because a directional light illuminates all objects in the scene from the same direction and with the same intensity, so the position of the node containing the light has no effect. Attenuation and spotlight angle attributes do not apply to directional lights.

13.26.8 SCNLightTypeIES as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A light source whose shape, direction, and intensity of illumination is determined by a photometric profile.
Notes: The intensity of a photometric light varies in different directions from the light source, much like the illumination from a real-world light source. The position of the containing node determines the location of the light source, and the orientation of the node determines the relative directions specified by the photometric profile. Spotlight angle attributes do not apply to photometric lights.
For more information about photometric lights, see the IESProfileURL property.

13.26.9 SCNLightTypeOmni as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: An omnidirectional light, also known as a point light.
Notes: Because an omnidirectional light casts equal illumination in all directions, the orientation of the node containing the light has no effect. Spotlight angle and shadow attributes do not apply to directional lights.

13.26.10 SCNLightTypeProbe as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A sample of the environment around a point in a scene to be used in environment-based lighting.
Notes: A light probe describes a point in a scene in terms of the variations in color and intensity of the illumination it receives from all directions. This information can then be used in shading of materials based on their location in the scene. For example, a white object placed near blue and red walls will appear bluish on surfaces facing the blue wall and reddish on surfaces facing the red wall.
You can place light probes in a scene and generate their lighting contributions using the Xcode scene editor, or import light probes from scene file formats that support them. Lighting-related properties of the SCNLight class do not apply to light probes; their contribution to scene rendering depends entirely on the light probe content generated in Xcode.

13.26.11 SCNLightTypeSpot as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A light source that illuminates a cone-shaped area.
Notes: The position and orientation of the node containing the light determines the area lit by the spotlight, and all lighting attributes affect its appearance.

13.26.12 Properties

13.26.13 attenuationEndDistance as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The distance from the light at which its intensity is completely diminished. Animatable.
Notes: You can apply attenuation to omnidirectional lights and spotlights, causing their intensity to diminish over a specified range of distances. At distances less than the start distance, the light,Äôs illumination is at full intensity. At distances greater than the end distance, the light provides no illumination. At distances in between the start and end distance, the attenuationFalloffExponent property defines the transition from full illumination to no illumination.
The default value is 0.0, specifying no attenuation (the light,Äôs intensity is the same at all distances).
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.14 attenuationFalloffExponent as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The transition curve for the light,Äôs intensity between its attenuation start and end distances. Animatable.
Notes: You can apply attenuation to omnidirectional lights and spotlights, causing their intensity to diminish over a specified range of distances. At distances in between the start and end distance, the light,Äôs intensity transitions from full to no illumination according to the value of this property.
A value of 0.0 specifies no attenuation—the light,Äôs intensity is the same at all distances. A value of 1.0 specifies a linear transition, and a value of 2.0 (the default) specifies a quadratic transition curve. Higher values have little visible effect.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.15 attenuationStartDistance as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The distance from the light at which its intensity begins to diminish. Animatable.
Notes: You can apply attenuation to omnidirectional lights and spotlights, causing their intensity to diminish over a specified range of distances. At distances less than the start distance, the light,Äôs illumination is at full intensity. At distances greater than the end distance, the light provides no illumination. At distances in between the start and end distance, the attenuationFalloffExponent property defines the transition from full illumination to no illumination.
The default value is 0.0, specifying no attenuation (the light,Äôs intensity is the same at all distances).
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.16 automaticallyAdjustsShadowProjection as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies if the shadow map projection should be done automatically or manually by the user.
Notes: Defaults to true.
(Read and Write property)

13.26.17 CastsShadow as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the light casts shadows.
Notes: Geometries illuminated by the light cast shadows only if the value of this property is true and the type property of the light is SCNLightTypeSpot or SCNLightTypeDirectional. The default value is false.
(Read and Write property)

13.26.18 Color as NSColorMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The color of the light. Animatable.
Notes: The value of this property is an NSColorMBS object. The default color is white.
You can animate changes to this property,Äôs value.

See Animating SceneKit Content:
https://developer.apple.com/documentation/scenekit/animation/animating_scenekit_content
(Read and Write property)

13.26.19 ForcesBackFaceCasters as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.26.20 gobo as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An image or other visual content affecting the shape and color of a light,Äôs illuminated area.
Notes: In photographic and stage lighting terminology, a gobo (also known as a flag or cookie) is a stencil, gel, or other object placed just in front of a light source, shaping or coloring the beam of light.
You alter the appearance of a spotlight by changing the contents property of the object permanently assigned to this property. As with other material properties, you can use a color or image, or a Core Animation layer containing animated content, as a lighting gobo.
This property applies only to lights whose type property is spot.
(Read only property)

13.26.21 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.26.22 Intensity as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The luminous flux, in lumens, or total brightness of the light. Animatable.
Notes: When working with photometric lights (see the IESProfileURL property) or physically-based rendering (see SCNLightingModelPhysicallyBased), you can leave the color property at its default white color and use the intensity and temperature to control the light using realistic parameters. When working with physically-based materials, this value the luminous flux of the light source. The default value is 1000 lumens. When not using physically-based rendering, this value (divided by 1000) serves as a multiplier for the the color property. The default value of of 1000 leaves the light color unmodulated; you can use higher values, for example, to brighten a light whose color is already the maximum red value.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.23 maximumShadowDistance as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the maximum distance from the viewpoint from which the shadows for the receiver light won't be computed.
Notes: Defaults to 100.0.
(Read and Write property)

13.26.24 Name as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A name associated with the light.
Notes: You can provide a descriptive name for a light to make managing your scene graph easier. Lights loaded from a scene file may have names assigned by an artist using a 3D authoring tool. To examine lights in a scene file without loading its scene graph, use the SCNSceneSourceMBS class.
(Read and Write property)

13.26.25 OrthographicScale as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The orthographic scale SceneKit uses when rendering the shadow map for a directional light.
Notes: SceneKit draws a shadow map image by rendering the scene from the point of view of the node containing the light. Directional lights ignore the position property of the node containing them because their light has a constant direction. Therefore, rendering a shadow map for a directional light requires an orthographic projection. Like the orthographicScale property of a camera object, this property specifies the extent of the scene ,Äúvisible to,Äù the light when rendering the shadow map.
This property applies only if the light,Äôs type property is SCNLightTypeDirectional.
(Read and Write property)

13.26.26 SampleDistributedShadowMaps as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Use the sample distribution of the main rendering to better fit the shadow frusta.
Notes: Defaults to false.
(Read and Write property)

13.26.27 ShadowBias as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The amount of correction to apply to the shadow to prevent rendering artifacts.
Notes: Rendering shadows from a shadow map can result in artifacts where the shadow color does not appear on all pixels in a shadowed surface as intended. (This effect is typically called shadow acne.) This property specifies the error margin SceneKit uses to correct such artifacts.
The default value is 1.0.
(Read and Write property)

13.26.28 ShadowCascadeCount as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the number of distinct shadow maps that will be computed for the receiver light.
Notes: Defaults to 1. Maximum is 4.
(Read and Write property)

13.26.29 ShadowCascadeSplittingFactor as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies a factor to interpolate between linear splitting (0) and logarithmic splitting (1).
Notes: Defaults to 0.15.
(Read and Write property)

13.26.30 shadowColor as NSColorMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The color of shadows cast by the light. Animatable.
Notes: The value of this property is an NSColorMBS object. SceneKit blends the light,Äôs color with other colors in the rendered image to produce a shadow effect. The color,Äôs opacity (alpha value) determines the intensity of the shadows. The default shadow color is black with 50% opacity.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.31 ShadowMode as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The mode SceneKit uses to render shadows.
Notes: The default mode is SCNShadowModeForward in iOS and in macOS 10.10 or later. In OS X v10.9 or earlier, the default mode is SCNShadowModeDeferred.
(Read and Write property)

13.26.32 ShadowRadius as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A number that specifies the amount of blurring around the edges of shadows cast by the light. Animatable.

Notes: SceneKit produces soft-edged shadows by rendering the silhouettes of geometry into a 2D shadow map and then using several weighted samples from the shadow map to determine the strength of the shadow at each pixel in the rendered scene. This property controls the radius of shadow map sampling. Lower numbers result in shadows with sharply defined, pixelated edges; higher numbers result in blurry shadows. You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.33 ShadowSampleCount as Integer

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The number of samples from the shadow map that SceneKit uses to render each pixel.
Notes: SceneKit produces soft-edged shadows by rendering the silhouettes of scene geometry into a 2D shadow map and then using several weighted samples from the shadow map to determine the strength of the shadow at each pixel in the rendered scene. This property controls the number of samples from the shadow map used to render each pixel. Higher numbers result in smoother edges; lower numbers increase rendering performance.
The default value is 16 in macOS and 1 on iOS.
(Read and Write property)

13.26.34 spotInnerAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The angle, in degrees, of the area fully lit by a spotlight. Animatable.
Notes: You define the cone-shaped illuminated area of a spotlight with a position and direction (from the node containing the light) and an angle specifying the cone,Äôs width. Additionally, the illuminated area can smoothly transition from full illumination to no illumination. This property determines the width of the fully illuminated area.
The default value is 0.0, specifying that only the center of the area illuminated by the spotlight is lit at full intensity.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.35 spotOuterAngle as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The angle, in degrees, of the area partially lit by a spotlight. Animatable.
Notes: You define the cone-shaped illuminated area of a spotlight with a position and direction (from the node containing the light) and with an angle specifying the cone,Äôs width. Additionally, the illuminated area can smoothly transition from full illumination to no illumination. This property determines the width of the transition area.

The default value is 45.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.36 Temperature as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The color temperature, in degrees Kelvin, of the light source. Animatable.
Notes: SceneKit determines the actual color of the light by multiplying the color value by a color corresponding to the light,Äôs temperature. The default value of 6500 K represents a pure white light (leaving the color unmodulated); lower values (down to a minimum of zero) add a ,Äúwarmer,Äù yellow or orange effect to the light source, and higher values (up to a maximum of 40000) add a ,Äúcooler,Äù blue effect. This property affects all light types, but is especially useful when working with photometric lights (see the IESProfileURL property) or physically-based rendering (see SCNLightingModelPhysicallyBased). You can leave the color property at its default white color and use the intensity and temperature properties to control the light using realistic parameters.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.37 Type as String

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: A constant identifying the general behavior of the light.
Notes: A light,Äôs type determines the shape and directionality of illumination provided by the light, as well as the set of attributes available for modifying the light,Äôs behavior. For example, light types include omnidirectional lights and spotlights. See Light Types for the full set of types and their behaviors.

see also
https://developer.apple.com/documentation/scenekit/scnlighttype
(Read and Write property)

13.26.38 zFar as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum distance between the light and a visible surface for casting shadows.
Notes: A spotlight casts shadows if its castsShadow property is true. If a surface is farther from the light than this distance, shadows are not cast against the surface.
The default value is 100.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.

(Read and Write property)

13.26.39 zNear as Double

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum distance between the light and a visible surface for casting shadows. Animatable.
Notes: A spotlight casts shadows if its castsShadow property is true. If a surface is closer to the light than this distance, shadows are not cast against the surface.
The default value is 1.0.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.26.40 Constants

Shadow Modes

Constant	Value	Description
kShadowModeDeferred	1	SceneKit renders shadows in a postprocessing pass. In the mode, SceneKit blends shadows into the final image after the main rendering pass, so shadows can be of any color.
kShadowModeForward	0	SceneKit renders shadows during lighting computations. In this mode, the color components of the light,Äôs shadowColor property do not apply. The color,Äôs alpha component determines the intensity of shadows.
kShadowModeModulated	2	SceneKit renders shadows by projecting the light,Äôs gobo image. The light does not illuminate the scene. Typically, you use this mode to create a low-accuracy, high-performance shadow under a game character or similar scene element: Use an image of a radial gradient (black to white) for the light,Äôs gobo property, and use categoryBitMask properties to prevent the shadow image from appearing on the character.

13.27 class SCNLookAtConstraintMBS

13.27.1 class SCNLookAtConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNLookAtConstraint applies on a node's orientation so that it always look at another node.
Notes: Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.27.2 Methods

13.27.3 Constructor(target as SCNNodeMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNLookAtConstraint object with the specified target.
Notes: target: The target node to look at.

13.27.4 lookAtConstraintWithTarget(target as SCNNodeMBS) as SCNLookAtConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNLookAtConstraint object with the specified target.
Notes: target: The target node to look at.

13.27.5 Properties

13.27.6 gimbalLockEnabled as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether the receiver enables the gimbal lock. Defaults to false.
Notes: Enabling the gimbal lock prevents the receiver from rotating the constrained node around to roll axis.
(Read and Write property)

13.27.7 localFront as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Front direction in the constraint owner local space. Defaults to - [SCNNode localFront] . Animatable
Notes: (Read and Write property)

13.27.8 target as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the target node to look at.
Notes: (Read and Write property)

13.27.9 targetOffset as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Offset look at position in target space. Defaults to zero. Animatable
Notes: (Read and Write property)

13.27.10 worldUp as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Up reference direction in world space. Defaults to - [SCNNode localUp] . Animatable
Notes: (Read and Write property)

13.28 class SCNMaterialMBS

13.28.1 class SCNMaterialMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A set of shading attributes that define the appearance of a geometry's surface when rendered.
Notes: When you create a material, you define a collection of visual attributes and their options, which you can then reuse for multiple geometries in a scene.

A material has several visual properties, each of which defines a different part of SceneKit,Äôs lighting and shading process. Each visual property is an instance of the SCNMaterialPropertyMBS class that provides a solid color, texture, or other 2D content for that aspect of SceneKit,Äôs rendering. The material,Äôs lightingModelName property then determines the formula SceneKit uses to combine the visual properties with the lights in the scene to produce the final color for each pixel in the rendered scene. For more details on the rendering process, see SCNLightingModelMBS.

You attach one or more materials to an instance of the SCNGeometryMBS class using its firstMaterial or materials property. Multiple geometries can reference the same material. In this case, changing the attributes of the material changes the appearance of every geometry that uses it.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.28.2 Methods

13.28.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material.

13.28.4 copy as SCNMaterialMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the material.

13.28.5 material as SCNMaterialMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material object.
Notes: A newly created material contains an SCNMaterialPropertyMBS object for each of its eight visual properties. You change a material,Äôs appearance by setting the contents of each visual property. For information on each visual property,Äôs default contents and how it affects a material,Äôs appearance, see Visual Properties for Special Effects.

13.28.6 SCNLightingModelBlinn as string

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Shading that incorporates ambient, diffuse, and specular properties, where specular highlights are calculated using the Blinn-Phong formula.
Notes: see
https://developer.apple.com/documentation/scenekit/scnlightingmodelblinn

13.28.7 SCNLightingModelConstant as string

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Uniform shading that incorporates ambient lighting only.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnlightingmodelconstant

13.28.8 SCNLightingModelLambert as string

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Shading that incorporates ambient and diffuse properties only.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnlightingmodellambert

13.28.9 SCNLightingModelPhong as string

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Shading that incorporates ambient, diffuse, and specular properties, where specular highlights are calculated using the Phong formula.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnlightingmodelphong

13.28.10 SCNLightingModelPhysicallyBased as string

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Shading based on a realistic abstraction of physical lights and materials.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnlightingmodelphysicallybased

13.28.11 Properties

13.28.12 ambient as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that manages the material,Äôs response to ambient lighting.
Notes: The ambient property specifies the amount of ambient light to reflect. This property has no visual impact on scenes that have no ambient light. Setting the ambient has no effect if locksAmbientWithDiffuse is set to true.
(Read only property)

13.28.13 ambientOcclusion as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that provides color values to be multiplied with the ambient light affecting the material.
Notes: Use this property to assign an ambient occlusion texture map to a surface. This property has no effect if there is no ambient light in the scene. If this property is not nil, SceneKit ignores the ambient property.
When using physically-based shading (see SCNLightingModelPhysicallyBased), ambient occlusion approximates large-scale surface details that obscure global illumination.
(Read only property)

13.28.14 blendMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The mode that determines how pixel colors rendered using this material blend with other pixel colors in the rendering target.
Notes: With the default blend mode of SCNBlendModeAlpha, materials blend according to their alpha (opacity) values—a pixel rendered with a higher alpha value appears more opaque than one with a lower alpha value. Change this property to create special effects. For example, the SCNBlendModeAdd mode can make objects appear to glow.

(Read and Write property)

13.28.15 colorBufferWriteMask as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether the receiver writes to the color buffer when rendered.
Notes: Defaults to SCNColorMaskAll.
(Read and Write property)

13.28.16 cullMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The mode determining which faces of a surface SceneKit renders. Animatable.
Notes: The vertex data and normal vectors in a geometry designate which side of each polygon is to be considered its front face, and the geometry,Äôs orientation with respect to the camera determines which front surfaces are currently visible. Typically, back-facing surfaces are found only on the interior of a closed geometry, obscured by front-facing surfaces, so rendering these surfaces has a performance cost but no visible effect.
This property,Äôs default value is SCNCullBack, specifying that SceneKit should cull, or not render, back-facing surfaces. You can change this property,Äôs value to cause SceneKit to render only the back surfaces of a material instead. See SCNCullMode for available values.
You can animate changes to this property,Äôs value. See Animating SceneKit Content. Animating this property fades between the results of rendering with each state
(Read and Write property)

13.28.17 diffuse as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that manages the material,Äôs diffuse response to lighting.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnmaterial/1462589-diffuse
(Read only property)

13.28.18 displacement as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The displacement property specifies how vertex are translated in tangent space.

Notes: Pass a grayscale image for a simple 'elevation' or rgb image for a vector displacement.
(Read only property)

13.28.19 DoubleSided as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit should render both front and back faces of a surface. Animatable.
Notes: When this property,Äôs value is false (the default), SceneKit renders a surface using the material from one side. The vertex data and normal vectors in a geometry designate which side of each polygon is to be considered its front face, and the geometry,Äôs orientation with respect to the camera determines which front surfaces are currently visible.
If you change this property,Äôs value to true, SceneKit renders both the front and back surfaces of every polygon. SceneKit assumes the surface normals a back face to be the negative of the front face,Äôs normal vector.
You can animate changes to this property,Äôs value. See Animating SceneKit Content. Animating this property fades between the results of rendering with each state
(Read and Write property)

13.28.20 emission as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that defines the color emitted by each point on a surface.
Notes: The emission property specifies the amount of light the material emits. This emission does not light up other surfaces in the scene.
(Read only property)

13.28.21 fillMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines of to how to rasterize the receiver's primitives.
Notes: Defaults to SCNFillModeFill.
(Read and Write property)

13.28.22 fresnelExponent as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A factor affecting the material,Äôs reflectivity. Animatable.
Notes: The Fresnel exponent of a material interacts with its reflective property to determine the intensity of reflections in a surface based on its angle relative to the viewer. A higher Fresnel exponent increases the visibility of reflections when the material is viewed from a shallow angle.
(Read and Write property)

13.28.23 lightingModelName as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The lighting formula that SceneKit uses to render the material.
Notes: SceneKit provides several different lighting models, each of which combines information from a material,Äôs visual properties with the lights and other contents of a scene. For details on how each lighting model affects rendering, see Lighting Models. For details on the contribution from each visual property, see Visual Properties for Special Effects.
(Read and Write property)

13.28.24 LitPerPixel as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit performs lighting calculations per vertex or per pixel. Animatable.
Notes: When this property,Äôs value is true (the default), SceneKit performs lighting calculations independently for each rendered pixel. This approach provides better rendering quality, but can adversely impact rendering performance.
If you change this property,Äôs value to false, SceneKit performs lighting calculations for each vertex in a geometry, and allows the GPU to interpolate lighting results across the pixels in between vertices. Depending on the shape and vertex count of a geometry,Äôs surface and the material properties being rendered, this approach may improve rendering performance without much noticeable impact on visual quality.
You can animate changes to this property,Äôs value. See Animating SceneKit Content. Animating this property fades between the results of rendering with each state.
(Read and Write property)

13.28.25 locksAmbientWithDiffuse as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the material responds identically to both ambient and diffuse lighting. Animatable.
Notes: Makes the ambient property automatically match the diffuse property. Defaults to false on 10.9 and before, defaults to true otherwise. Animatable.

(Read and Write property)

13.28.26 metalness as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that provides color values to determine how metallic the material,Äôs surface appears.
Notes: This property measures only the total intensity of color values; texture contents are best defined in grayscale.
This property generally approximates aspects of a physical surface—such as index of refraction, tendency to produce sharp reflections, and tendency to produce Fresnel reflections at grazing angles—that together produce an overall metallic or nonmetallic (also called dielectric) appearance. Lower values (darker colors) cause the material to appear more like a dielectric surface. Higher values (brighter colors) cause the surface to appear more metallic.
This property applies only when the material,Äôs lightingModelName value is SCNLightingModelPhysically-Based.
(Read only property)

13.28.27 multiply as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that provides color values that are multiplied with pixels in a material after all other shading is complete.
Notes: The multiply property specifies a color or an image used to multiply the output fragments with. The computed fragments are multiplied with the multiply value to produce the final fragments. This property may be used for shadow maps, to fade out or tint 3d objects.
(Read only property)

13.28.28 name as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A name associated with the material.
Notes: You can provide a descriptive name for a material to make managing your scene graph easier. Materials loaded from a scene file may have names assigned by an artist using a 3D authoring tool. Use the SCNSceneSourceMBS class to examine materials in a scene file without loading its scene graph.

Material names are saved when you export a scene to a file using its writeToURL method, and appear in the Xcode scene editor.
(Read and Write property)

13.28.29 normal as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that defines the nominal orientation of the surface at each point for use in lighting.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnmaterial/1462542-normal
(Read only property)

13.28.30 readsFromDepthBuffer as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether the receiver reads from the depth buffer when rendered.
Notes: Defaults to true.
(Read and Write property)

13.28.31 reflective as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that defines the reflected color for each point on a surface.
Notes: The reflective property specifies the reflectivity of the surface. The surface will not actually reflect other objects in the scene. This property may be used as a sphere mapping to reflect a precomputed environment.
(Read only property)

13.28.32 roughness as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that provides color values to determine the apparent smoothness of the surface.
Notes: This property measures only the total intensity of color values; texture contents are best defined in grayscale.
This property approximates the level of microscopic detail—for example tiny bumps and cracks—in a surface. By approximating these ‚Äúmicrofacets‚Äù as a single term, this property helps produce lighting calculations that resemble the energy-conserving laws of real-world physics, resulting in more realistic variation between matte and shiny surfaces. Lower values (darker colors) cause the material to appear shiny, with well-defined specular highlights. Higher values (brighter colors) cause specular highlights to spread out and the diffuse color of the material to become more retroreflective.
This property applies only when the material‚Äôs lightingModelName value is SCNLightingModelPhysicallyBased.

(Read only property)

13.28.33 selfIllumination as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that provides color values representing the global illumination of the surface.
Notes: Self-illumination applies to all materials, but is especially useful for those using physically-based shading (see SCNLightingModelPhysicallyBased). Physically-based materials work best with environment-based lighting (see the SCNSceneMBS property lightingEnvironment), but for some materials it can be useful to let a surface itself define part of its lighting—for example, an object whose position obscures it from the ‚Äúsky‚Äù that provides the main lighting environment. When you assign contents to this property, they override the environmental lighting contribution to diffuse shading, but environmental lighting still contributes to specular effects.
(Read only property)

13.28.34 shininess as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The sharpness of specular highlights. Animatable.
Notes: The shininess of a material interacts with its specular property and the lighting in a scene to produce bright highlights on a surface. A higher value produces more sharply defined highlights, making a surface appear more smooth and glossy.
(Read and Write property)

13.28.35 specular as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that manages the material‚Äôs specular response to lighting.
Notes: The specular property specifies the amount of light to reflect in a mirror-like manner. The specular intensity increases when the point of view lines up with the direction of the reflected light.
(Read only property)

13.28.36 transparency as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The uniform transparency of the material. Animatable.
Notes: SceneKit determines the total opacity of each rendered pixel in a surface by multiplying the color

from the material,Äôs transparent property by the value of this property. Then, the material,Äôs transparencyMode property determines how pixels from the material are blended into the scene.
You can also uniformly adjust the opacity of all content attached to a node using its opacity property.
(Read and Write property)

13.28.37 transparencyMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The mode SceneKit uses to calculate transparency for the material.
Notes: The default transparency mode is SCNTransparencyModeAOne. See SCNTransparencyMode for available values and their effects.
(Read and Write property)

13.28.38 transparent as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that determines the opacity of each point in a material.
Notes: The transparent property specifies the transparent areas of the material.
(Read only property)

13.28.39 writesToDepthBuffer as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit produces depth information when rendering the material.
Notes: Determines whether the receiver writes to the depth buffer when rendered. Defaults to true.
(Read and Write property)

13.28.40 Constants

Constants

Constant	Value	Description
SCNTransparencyModeAOne	0	One of the constants for the modes SceneKit uses to calculate the opacity pixels rendered with a material, used by the transparencyMode property. SceneKit derives transparency information from the alpha channel of colo The value 1.0 is opaque.
SCNTransparencyModeDefault	0	One of the constants for the modes SceneKit uses to calculate the opacity pixels rendered with a material, used by the transparencyMode property. Default
SCNTransparencyModeDualLayer	3	One of the constants for the modes SceneKit uses to calculate the opacity pixels rendered with a material, used by the transparencyMode property. Dual layer.
SCNTransparencyModeRGBZero	1	One of the constants for the modes SceneKit uses to calculate the opacity pixels rendered with a material, used by the transparencyMode property. SceneKit derives transparency information from the luminance of colors. T value 0.0 is opaque. When using this mode, SceneKit ignores the alpha value of colors in the m terial,Äôs transparent property. SceneKit calculates the luminance of a co from its red, green, and blue channels and uses the resulting value to determi the material,Äôs opacity.
SCNTransparencyModeSingleLayer	2	One of the constants for the modes SceneKit uses to calculate the opacity pixels rendered with a material, used by the transparencyMode property. Single layer.

Blend Modes

Constant	Value	Description
SCNBlendModeAdd	1	Blend by adding the source color to the destination color. This mode results in a brightening effect that can be useful for making objects appear to glow relative to their surroundings.
SCNBlendModeAlpha	0	Blend by multiplying source and destination color values by their corresponding alpha values. This mode is the default value of the blendMode property.
SCNBlendModeMax	6	Max
SCNBlendModeMultiply	3	Blend by multiplying the source color with the background color. This mode results in colors that are at least as dark as either of the two contributing colors.
SCNBlendModeReplace	5	Blend by replacing the destination color with the source color, ignoring alpha.
SCNBlendModeScreen	4	Blend by multiplying the inverse of the source color with the inverse of the destination color. This mode results in colors that are at least as light as either of the two contributing colors.
SCNBlendModeSubtract	2	Blend by subtracting the source color from the destination color.

Cull Modes

Constant	Value	Description
SCNCullModeBack	0	The mode for culling back-facing polygons (and rendering only front-facing polygons).
SCNCullModeFront	1	The mode for culling front-facing polygons (and rendering only back-facing polygons).

Fill Modes

Constant	Value	Description
SCNFillModeFill	0	Fill
SCNFillModeLines	1	Lines

13.29 class SCNMaterialPropertyMBS

13.29.1 class SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A container for the color or texture of one of a material,Äôs visual properties.
Notes: A material has several visual properties that together determine its appearance under lighting and shading. SceneKit renders each pixel in the scene by combining the information from material properties with the locations, intensities, and colors of lights.
A material property,Äôs contents can be either a color, which provides a uniform effect across the surface of a material, or a texture, which SceneKit maps across the surface of a material using texture coordinates provided by the geometry object the material is attached to. A texture, in turn, can come from any of several sources, such as an image object, a URL to an image file, a specially formatted image or set of images for use as a cube map, or even animated content provided by Core Animation, SpriteKit, or AVFoundation—for the full set of options, see the contents property.

see also
https://developer.apple.com/documentation/scenekit/scnmaterialproperty
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.29.2 Methods

13.29.3 Constructor(content as Variant)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property with given content.

13.29.4 materialPropertyWithColor(color as NSColorMBS) as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property object with the specified color.

13.29.5 materialPropertyWithContents(content as Variant) as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property object with the specified contents.
Notes: contents: The visual contents of the material property—a color, image, or source of animated content. For details, see the discussion of the contents property.

Returns a new material property object.

Newly created SCNMaterialMBS objects contain SCNMaterialPropertyMBS instances for all of their visual properties. To change a material,Äôs visual properties, you modify those instances rather than creating new material property objects.
You create new SCNMaterialPropertyMBS instances to provide textures for use with custom GLSL shaders—for details, see SCNShadableMBS.

13.29.6 materialPropertyWithImage(image as NSImageMBS) as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property object with the specified image.

13.29.7 materialPropertyWithLayer(layer as CALayerMBS) as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property object with the specified layer.

13.29.8 materialPropertyWithPicture(picture as Picture) as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new material property object with the specified picture.

13.29.9 Properties

13.29.10 borderColor as NSColorMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A color used to fill in areas of a material,Äôs surface not covered by the material property,Äôs image contents.
Notes: When the material property,Äôs contents are a texture image and its texture wrapping properties are set to SCNWrapModeClampToBorder, the border color appears in areas of a textured geometry not covered by the texture image, as shown in .
(Read and Write property)

13.29.11 contents as Variant

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The visual contents of the material property—a color, image, or source of animated content. Animatable.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnmaterialproperty/1395372-contents
(Read and Write property)

13.29.12 contentsTransform as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The transformation applied to the material property,Äôs visual contents. Animatable.
Notes: SceneKit applies this transformation to the texture coordinates provided by the geometry object the material is attached to, then uses the resulting coordinates to map the material property,Äôs contents across the surface of the material. (This transformation has no effect if the material property,Äôs contents object is a constant color.)
(Read and Write property)

13.29.13 intensity as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A number between 0.0 and 1.0 that modulates the effect of the material property. Animatable.
Notes: The default intensity is 1.0. Reducing the intensity fades out the contents of the material property, causing different effects depending on which visual property of an SCNMaterialMBS object it represents:

- For the normal property, intensity varies the apparent roughness of the normal-mapped surface. Reducing intensity makes the surface appear more smooth.
- For the multiply property, reducing intensity blends the material property,Äôs colors with white, effectively reducing the strength of the color multiplication effect.
- For all other properties, reducing intensity dims the material property,Äôs contents.

(Read and Write property)

13.29.14 magnificationFilter as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Texture filtering for rendering the material property,Äôs image contents at a size larger than that of the original image.
Notes: Texture filtering determines the appearance of a material property,Äôs contents when portions of the material surface appear larger or smaller than the original texture image. For example, the texture coordinates at a point near the camera may correspond to a small fraction of a pixel in the texture image. SceneKit uses the magnification filter to determine the color of the sampled texel at that point.
The default magnification filter is SCNFilterModeLinear.
(Read and Write property)

13.29.15 mappingChannel as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The source of texture coordinates for mapping the material property,Äôs image contents.
Notes: A geometry can have multiple independent sources of texture coordinates, each of which defines a unique mapping channel number. You can use these channels to map different visual properties of a material in different ways. For example, a geometry representing a picture frame might use one set of texture coordinates for mapping the materials of the frame itself, and another set of texture coordinates for placing a picture into the frame.
For information about creating geometries with multiple texture mapping channels, see SCNGeometryMBS.
(Read and Write property)

13.29.16 maxAnisotropy as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The amount of anisotropic texture filtering to be used when rendering the material property,Äôs image contents.
Notes: see
https://developer.apple.com/documentation/scenekit/scnmaterialproperty/1395402-maxanisotropy
(Read and Write property)

13.29.17 minificationFilter as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Texture filtering for rendering the material property,Äôs image contents at a size smaller than that of the original image.
Notes: Texture filtering determines the appearance of a material property,Äôs contents when portions of the material surface appear larger or smaller than the original texture image. For example, the texture coordinates at a point far from the camera may correspond to an area of several pixels in the texture image. SceneKit uses the minification filter to determine the color of the sampled texel at that point.
The default minification filter is SCNFilterModeLinear. See constants for available modes and their effects.
(Read and Write property)

13.29.18 mipFilter as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Texture filtering for using mipmaps to render the material property,Äôs image contents at a size smaller than that of the original image.
Notes: see
https://developer.apple.com/documentation/scenekit/scnmaterialproperty/1395398-mipfilter
(Read and Write property)

13.29.19 wrapS as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The wrapping behavior for the S texture coordinate.
Notes: Wrapping modes determine texture mapping behavior for cases where a material,Äôs texture coordinates extend outside the range from 0.0 to 1.0. For example, if you use the contentsTransform property to shrink a texture relative to the surface of a geometry, you use the wrap mode properties to determine whether the texture repeats across the surface.
The S texture coordinate measures the horizontal axis of a texture image, increasing from 0.0 at the left edge of the image to 1.0 at the right edge.
The default wrap mode is SCNWrapModeClamp. See constants for available modes and their effects.
(Read and Write property)

13.29.20 wrapT as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The wrapping behavior for the T texture coordinate.
Notes: Wrapping modes determine texture mapping behavior for cases where a material,Äôs texture coordinates extend outside the range from 0.0 to 1.0. For example, if you use the contentsTransform property to shrink a texture relative to the surface of a geometry, you use the wrap mode properties to determine whether the texture repeats across the surface.
The T texture coordinate measures the vertical axis of a texture image, increasing from 0.0 at the bottom

of the image to 1.0 at the top.
The default wrap mode is SCNWrapModeClamp.
(Read and Write property)

13.29.21 Constants

Filter Modes

Constant	Value	Description
SCNFilterModeLinear	2	Texture filtering sample texels from the neighborhood of the coordinates being sampled and linearly interpolates their colors.
SCNFilterModeNearest	1	Texture filtering returns the color from only one texel, whose location is nearest to the coordinates being sampled.
SCNFilterModeNone	0	No texture filtering is applied. Only valid for the mipFilter property, specifying that SceneKit should not use mip mapping.

Wrap Modes

Constant	Value	Description
SCNWrapModeClamp	1	Texture coordinates are clamped to the range from 0.0 to 1.0, inclusive. Texture sampling in areas whose texture coordinates would fall outside this range produces texel colors from the nearest edge of the texture image.
SCNWrapModeClampToBorder	3	Texture sampling uses texture colors for coordinates in the range from 0.0 to 1.0 (inclusive) and the material property,Äôs borderColor value otherwise. Texture sampling in areas whose texture coordinates would fall outside this range uses the borderColor property instead of texel colors from the texture image.
SCNWrapModeMirror	4	Texture sampling of texture coordinates outside range from 0.0 to 1.0 should behave as if the range reverses before repeating. Texture sampling in areas of the material whose texture coordinates would fall outside from 0.0 to 1.0 results in tiling both texture image and its mirror image across the surface using the material.
SCNWrapModeRepeat	2	Texture sampling uses only the fractional part of texture coordinates, passing through the range from 0.0 to (but not including) 1.0. Texture sampling in areas of the material whose texture coordinates would fall outside from 0.0 to 1.0 results in tiling the texture image across the surface using the material.

13.30 class SCNMatrix4MBS

13.30.1 class SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A representation of a 4 x 4 matrix.
Notes: SceneKit uses matrices to represent coordinate space transformations, which in turn can represent the combined position, rotation or orientation, and scale of an object in three-dimensional space.
Blog Entries

- MBS Xojo Plugins, version 23.5pr6
- MBS Xojo Plugins, version 19.1pr1

13.30.2 Methods

13.30.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.
Notes: Initializes all values with zero.

13.30.4 copy as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the matrix.

13.30.5 equals(other as SCNMatrix4MBS) as boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a Boolean value that indicates whether the corresponding elements of two matrices are equal.
Notes: self: The first matrix to be compared.
other: The first matrix to be compared.

Returns true if each element in self is exactly equal to the corresponding element in other.

This function performs a numeric (not bitwise) comparison of each pair of elements.

13.30.6 Identity as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The 4 x 4 identity matrix.
Notes: Elements on the diagonal of this matrix are 1.0; all other elements are 0.0. Multiplying another matrix by the identity matrix or multiplying the identity matrix by another matrix yields the other matrix.

13.30.7 Invert as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the inverse of the specified matrix.
Notes: The inverse matrix of the specified matrix, or the original matrix if it is not invertible.

13.30.8 Multiply(other as SCNMatrix4MBS) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the product of two matrices.
Notes: self: The multiplicand, or left operand of matrix multiplication.
other: The multiplier, or right operand of matrix multiplication.

Returns the matrix product of the self and other parameters.

Matrix multiplication is not commutative. As a transformation, the result of multiplying a matrix A by a matrix B is the transformation represented by B followed by the transformation represented by A.

13.30.9 Null as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The null matrix.
Notes: All values are zero.

13.30.10 Rotate(angle as double, sx as double, sy as double, sz as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new matrix created by concatenating the specified matrix with a rotation transformation.
Notes: mat: The matrix to be combined with a rotation.
angle: The amount of rotation, in radians, measured counterclockwise around the rotation axis.
sx: The x-component of the rotation axis.
sy: The y-component of the rotation axis.
sz: The z-component of the rotation axis.

Returns a new matrix.

The resulting transformation consists of the specified rotation followed by the transformation represented by the mat parameter.

13.30.11 Rotation(angle as double, x as double, y as double, z as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a matrix describing a rotation transformation.
Notes: angle: The amount of rotation, in radians, measured counterclockwise around the rotation axis.
x: The x-component of the rotation axis.
y: The y-component of the rotation axis.
z: The z-component of the rotation axis.

Returns a new rotation matrix.

13.30.12 Scale(sx as double, sy as double, sz as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new matrix created by concatenating the specified matrix with a scale transformation.
Example:

```
Dim i As SCNMatrix4MBS = SCNMatrix4MBS.Identity
Dim m As SCNMatrix4MBS = i.Scale(1.0, 2.0, 3.0)
Break
```

Notes: self: The matrix to be combined with a translation.
sx: The scale factor in the x-axis direction.
sy: The scale factor in the y-axis direction.
sz: The scale factor in the z-axis direction.

Returns a new matrix.

The resulting transformation consists of the specified scale followed by the transformation represented by the mat parameter.

13.30.13 Scaling(sx as double, sy as double, sz as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a matrix describing a scale transformation.
Example:

```
Dim m As SCNMatrix4MBS = SCNMatrix4MBS.Scaling(1.0, 2.0, 3.0)
Break
```

Notes: sx: The scale factor in the x-axis direction.
sy: The scale factor in the y-axis direction.
sz: The scale factor in the z-axis direction.

Returns a new scale matrix.

13.30.14 Translate(tx as double, ty as double, tz as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new matrix created by concatenating the specified matrix with a translation transformation.
Notes: self: The matrix to be combined with a translation.
tx: The translation distance in the x-axis direction.
ty: The translation distance in the y-axis direction.
tz: The translation distance in the z-axis direction.

Returns a new matrix.

The resulting transformation consists of the specified translation followed by the transformation represented by the mat parameter.

13.30.15 Translation(tx as double, ty as double, tz as double) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a matrix describing a translation transformation.
Notes: tx: The translation distance in the x-axis direction.
ty: The translation distance in the y-axis direction.
tz: The translation distance in the z-axis direction.

Returns a new translation matrix.

13.30.16 Properties

13.30.17 IsIdentity as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a Boolean value that indicates whether the specified matrix is equal to the identity matrix.
Notes: Returns true if the elements on the matrix,Äôs diagonal are 1.0 and all other elements are 0.0.
(Read only property)

13.30.18 m11 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 1/1.
Notes: (Read and Write property)

13.30.19 m12 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 1/2.
Notes: (Read and Write property)

13.30.20 m13 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 1/3.
Notes: (Read and Write property)

13.30.21 m14 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 1/4.
Notes: (Read and Write property)

13.30.22 m21 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 2/1.
Notes: (Read and Write property)

13.30.23 m22 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 2/2.
Notes: (Read and Write property)

13.30.24 m23 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 2/3.
Notes: (Read and Write property)

13.30.25 m24 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 2/4.
Notes: (Read and Write property)

13.30.26 m31 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 3/1.
Notes: (Read and Write property)

13.30.27 m32 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 3/2.
Notes: (Read and Write property)

13.30.28 m33 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 3/3.
Notes: (Read and Write property)

13.30.29 m34 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 3/4.
Notes: (Read and Write property)

13.30.30 m41 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 4/1.
Notes: (Read and Write property)

13.30.31 m42 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 4/2.
Notes: (Read and Write property)

13.30.32 m43 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 4/3.
Notes: (Read and Write property)

13.30.33 m44 as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The value at 4/4.
Notes: (Read and Write property)

13.31 class SCNNodeMBS

13.31.1 class SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A structural element of a scene graph, representing a position and transform in a 3D coordinate space, to which you can attach geometry, lights, cameras, or other displayable content.
Notes: see also
https://developer.apple.com/documentation/scenekit/scnnode
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- News from the MBS Xojo Plugins Version 22.1
- Basic Forms in MBS Xojo SceneKit Plugin
- Physics example for SceneKit in Xojo
- MBS Xojo Plugins, version 20.2pr1
- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 19.6, page 50: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 19.6, pages 47 to 48: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, pages 48 to 50: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.4, pages 41 to 46: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 41: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 36 to 38: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 31 to 32: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 21: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, pages 32 to 33: On the Scene, Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 30: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.31.2 Methods

13.31.3 addAudioPlayer(player as SCNAudioPlayerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Adds the specified auto player to the node and begins playback.
Notes: player: An audio player object.

Positional audio effects from a player attached to a node are based on that node,Äôs position relative to the audioListener position in the scene.
After playback has completed, SceneKit automatically removes the audio player from the node.

You may need to have the scene/view in play mode to hear something.

13.31.4 addChildNode(node as SCNNodeMBS)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a node to the node,Äôs array of children.
Notes: child: The node to be added.

Calling this method appends the node to the end of the childNodes array.

13.31.5 audioPlayers as SCNAudioPlayerMBS()

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The audio players currently attached to the node.
Notes: Positional audio effects from a player attached to a node are based on that node,Äôs position relative to the audioListener position in the scene.
After an audio player completes playback, SceneKit automatically removes it from the node. Therefore, this array always contains audio players that are currently playing back audio.

13.31.6 childNodes as SCNNodeMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: An array of the node,Äôs children in the scene graph hierarchy.

13.31.7 childNodeWithName(Name as String, recursively as Boolean = false) as SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the first node in the node,Äôs child node subtree with the specified name.
Notes: name: The name of the node to search for.
recursively: True to search the entire child node subtree, or false to search only the node,Äôs immediate children.

If the recursive parameter is true, SceneKit uses a preorder traversal to search the child node subtree—that is, the block searches a node before it searches each of the node,Äôs children, and it searches all children of a node before searching any of that node,Äôs sibling nodes. Otherwise, SceneKit searches only those nodes in the node,Äôs childNodes array.

13.31.8 clearGeometry

Plugin Version: 22.1, Platform: macOS, Targets: Desktop & iOS.

Function: Clears geometry property.
Example:

```
if MyScene <>nil then
// clear geometry for all nodes in this Scene
MyScene.rootNode.clearGeometry
end if
```

Notes: Recursively walks over child nodes to also clear geometry by assigning nil.
Calling this method can help to mitigate problems where textures would not unload properly due to bugs in SceneKit framework.

13.31.9 clone as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the node and its children.
Notes: This method recursively copies the node and its child nodes. For a nonrecursive copy, use the inherited copy method, which creates a copy of the node without any child nodes.
Cloning or copying a node creates a duplicate of the node object, but not the geometries, lights, cameras, and other SceneKit objects attached to it—instead, each copied node shares references to these objects.

This behavior means that you can use cloning to, for example, place the same geometry at several locations

within a scene without maintaining multiple copies of the geometry and its materials. However, it also means that changes to the objects attached to one node will affect other nodes that share the same attachments. For example, to render two copies of a node using different materials, you must copy both the node and its geometry before assigning a new material.

Multiple copies of an SCNGeometryMBS object efficiently share the same vertex data, so you can copy geometries without a significant performance penalty.

13.31.10 constraints as SCNConstraintMBS()

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A list of constraints affecting the node,Äôs transformation.
Notes: An array of constraint objects. Before rendering, SceneKit evaluates all constraints attached to a node hierarchy and adjusts node transformations appropriately.
Use the SCNLookAtConstraintMBS class to make a node always point toward another node even as both are moved, or the SCNTransformConstraintMBS class to apply arbitrary transformations at constraint evaluation time.

13.31.11 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a new node object.
See also:

- 13.31.12 Constructor(geometry as SCNGeometryMBS) 1019

13.31.12 Constructor(geometry as SCNGeometryMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a node object with the specified geometry attached.
Notes: geometry: The geometry to be attached.

Returns a new node object with the geometry attached, or nil if initialization is not successful.
See also:

- 13.31.11 Constructor 1019

13.31.13 convertPositionFromNode(position as SCNVector3MBS, node as SCNNodeMBS) as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a position to the node,Äôs local coordinate space from that of another node.
Notes: position: A position in the local coordinate space defined by the other node.
node: Another node in the same scene graph as the node, or nil to convert from the scene,Äôs world coordinate space.

Returns a position in the node,Äôs local coordinate space.

13.31.14 convertPositionToNode(position as SCNVector3MBS, node as SCNNodeMBS) as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a position from the node,Äôs local coordinate space to that of another node.
Notes: position: A position in the node,Äôs local coordinate space.
node: Another node in the same scene graph as the node, or nil to convert to the scene,Äôs world coordinate space.

Returns a position in the local coordinate space defined by the other node.

13.31.15 convertTransformFromNode(position as SCNMatrix4MBS, node as SCNNodeMBS) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a transform to the node,Äôs local coordinate space from that of another node.
Notes: transform: A transform relative to the local coordinate space defined by the other node.
node: Another node in the same scene graph as the node, or nil to convert from the scene,Äôs world coordinate space.

Returns a transform relative to the node,Äôs coordinate space.

13.31.16 convertTransformToNode(position as SCNMatrix4MBS, node as SCNNodeMBS) as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a transform from the node,Äôs local coordinate space to that of another node.
Notes: transform: A transform relative to the node,Äôs coordinate space.
node: Another node in the same scene graph as the node, or nil to convert to the scene,Äôs world coordinate space.

Returns a transform relative to the local coordinate space defined by the other node.

13.31.17 convertVectorFromNode(position as SCNVector3MBS, node as SCNNodeMBS) as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a direction vector to the node,Äôs local coordinate space from that of another node.
Notes: vector: A direction vector in the local coordinate space defined by the other node.
node: Another node in the same scene graph as the node, or nil to convert from the scene,Äôs world coordinate space.

Returns a direction vector in the node,Äôs local coordinate space.

Unlike the convertPositionFromNode method, this method ignores the translational aspect of both nodes' transforms. As such, this method is more appropriate for use with vectors that represent only directional information, such as velocity or facing.

13.31.18 convertVectorToNode(position as SCNVector3MBS, node as SCNNodeMBS) as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a direction vector from the node,Äôs local coordinate space to that of another node.
Notes: vector: A direction vector in the node,Äôs local coordinate space.
node: Another node in the same scene graph as the node, or nil to convert to the scene,Äôs world coordinate space.

Returns a direction vector in the local coordinate space defined by the other node.

Unlike the convertPositionToNode method, this method ignores the translational aspect of both nodes' transforms. As such, this method is more appropriate for use with vectors that represent only directional information, such as velocity or facing.

13.31.19 copy as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the node.

13.31.20 flattenedClone as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an optimized copy of the node and its children.
Notes: A new single node containing the combined geometries and materials of the node and its child node subtree.

Rendering complex node hierarchies can incur a performance cost. Each geometry and material requires a separate draw command to be sent to the GPU, and each draw command comes with a performance overhead. If you plan for a portion of your scene,Äôs node hierarchy to remain static (with respect to itself, if not the rest of the scene), use this method to create a single node containing all elements of that node hierarchy that SceneKit can render using fewer draw commands.

13.31.21 hitTestWithSegment(PointA as SCNVector3MBS, PointB as SCNVector3MBS, Options as Dictionary = nil) as SCNHitTestResultMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Searches the node,Äôs child node subtree for objects intersecting a line segment between two specified points.
Notes: pointA: An endpoint of the line segment to search along, specified in the node,Äôs local coordinate system.
pointB: The other endpoint of the line segment to search along, specified in the node,Äôs local coordinate system.
options: A dictionary of options affecting the search. See Key shared methods in SCNHitTestResultMBS class for acceptable values.

Returns an array of SCNHitTestResultMBS objects representing search results.

Hit-testing is the process of finding elements of a scene located along a specified line segment in the scene,Äôs coordinate space (or that of a particular node in the scene). For example, you can use this method to determine whether a projectile launched by a game character will hit its target.

To search for the scene element corresponding to a two-dimensional point in the rendered image, use the renderer,Äôs hitTest method instead.

13.31.22 insertChildNode(newNode as SCNNodeMBS, atIndex as Integer)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a node to the node,Äôs array of children at a specified index.

13.31.23 localRotateBy(rotation as SCNVector4MBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Changes the node's orientation relative to its current orientation.
Notes: rotation: The axis and angle of rotation to apply, in node-local space, expressed as a quaternion.

This method rotates the node according to its pivot transform.
The effects of this method are animatable; that is, calling this method during an implicit-animation transaction animates the rotation effect.

Available in macOS 10.13 or later.

13.31.24 localTranslateBy(translation as SCNVector3MBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Changes the node's position relative to its current position.
Notes: translation: The distance, in node-local space, by which to move the node.

The effects of this method are animatable; that is, calling this method during an implicit-animation transaction animates the move.

Available in macOS 10.13 or later.

13.31.25 lookAt(worldTarget as SCNVector3MBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Changes the node's orientation so that its local forward vector points toward the specified location.

Notes: worldTarget: The point, in world space, to face the node toward.

Using this method is equivalent to calling the other lookAt method and passing the node's worldUp and localFront vectors for the corresponding parameters.
The effects of this method are animatable; that is, calling this method during an implicit-animation transaction animates the rotation effect.

Available in macOS 10.13 or later.
See also:

- 13.31.26 lookAt(worldTarget as SCNVector3MBS, worldUp as SCNVector3MBS, localFront as SCNVector3MBS) 1024

13.31.26 lookAt(worldTarget as SCNVector3MBS, worldUp as SCNVector3MBS, localFront as SCNVector3MBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Changes the node's orientation so that the specified forward vector points toward the specified location.
Notes: worldTarget: The point, in world space, to face the node toward.
worldUp: The direction vector, in world space, that should appear as "up" from the rotated node's point of view.
localFront: The direction vector, in the node's local space, that should orient toward the target point.

The effects of this method are animatable; that is, calling this method during an implicit-animation transaction animates the rotation effect.

Available in macOS 10.13 or later.
See also:

- 13.31.25 lookAt(worldTarget as SCNVector3MBS) 1023

13.31.27 node as SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a node object.
Notes: Returns a new node object, or nil if initialization is not successful.

13.31.28 nodeWithGeometry(geometry as SCNGeometryMBS) as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a node object with the specified geometry attached.
Notes: Returns a new node object with the geometry attached, or nil if initialization is not successful.

13.31.29 removeAllActions

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Ends and removes all actions from the node.
Notes: When SceneKit removes an action from a node, it skips any remaining animation the action would perform. However, any changes the action has already made to the node,Äôs state remain in effect.

13.31.30 removeAllAudioPlayers

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes all audio players attached to the node, stopping playback.

13.31.31 removeAudioPlayer(player as SCNAudioPlayerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the specified audio player from the node, stopping playback.
Notes: player: An audio player attached to the node.

This method has no effect if the player parameter does not reference an audio player directly attached to the node.

13.31.32 removeFromParentNode

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the node from its parent,Äôs array of child nodes.
Notes: Removing nodes from the node hierarchy serves two purposes. Nodes own their contents (child nodes or attached lights, geometries, and other objects), so deallocating unneeded nodes can reduce memory usage. Additionally, SceneKit does more work at rendering time with a large, complex node hierarchy, so removing nodes whose contents you don,Äôt need to display can improve rendering performance.

13.31.33 replaceChildNode(oldNode as SCNNodeMBS, newNode as SCNNodeMBS)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a child from the node,Äôs array of children and inserts another node in its place.
Notes: oldNode: The existing child node to be replaced.
newNode: The node with which to replace the child node.

If both the oldNode and newNode nodes are children of the node, calling this method swaps their positions in the array. Note that removing a node from the node hierarchy may result in it being deallocated.
Calling this method results in undefined behavior if the child parameter does not refer to a child of this node.

13.31.34 rotateBy(worldRotation as SCNVector4MBS, worldTarget as SCNVector3MBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Changes the node's position and orientation, relative to its current transform, through a rotation around the specified point in scene space.
Notes: worldRotation: The axis and angle of rotation to apply, in scene space, expressed as a quaternion.
worldTarget: The center point, in scene space, about which to rotate.

The effects of this method are animatable; that is, calling this method during an implicit-animation transaction animates the rotation effect.

Available in macOS 10.13 or later.

13.31.35 runAction(action as SCNActionMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an action to the list of actions executed by the node.
Notes: action: The action to be performed.

SceneKit begins running a newly added action when it prepares to render the next frame.
See also:

- 13.31.36 runAction(action as SCNActionMBS, del as SCNNodeRunActionCompletedMBS, tag as variant = nil) 1027
- 13.31.37 runAction(action as SCNActionMBS, key as string) 1027

13.31.36 runAction(action as SCNActionMBS, del as SCNNodeRunActionCompletedMBS, tag as variant = nil)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an action to the list of actions executed by the node. SceneKit calls the specified block when the action completes.
Notes: action: The action to be performed.
del: The method that SceneKit calls when the action completes.
tag: The tag is passed through to delegate method.

The new action is processed the next time SceneKit prepares to render a frame.
SceneKit calls your method after the action,Äôs duration is complete. For example, in a game you could use this method to show a Game Over message after performing a fade-out action on a node that displays a player character.
See also:

13.31.37 runAction(action as SCNActionMBS, key as string)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an identifiable action to the list of actions executed by the node.
Notes: action: The action to be performed.
key: A unique key used to identify the action.

This method is identical to runAction, but the action is stored and identified so that you can retrieve or cancel it later. If an action using the same key is already running, SceneKit removes it before adding the new action.
See also:

13.31.38 runAction(action as SCNActionMBS, key as string, del as SCNNodeRunActionCompletedMBS, tag as variant = nil)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an identifiable action to the list of actions executed by the node.
Notes: SceneKit calls the specified block when the action completes.

action: The action to be performed.
key: A unique key used to identify the action.
del: A completion block called when the action completes.
tag: The tag is passed through to delegate method.

This method is identical to runAction, but the action is stored and identified so that you can retrieve or cancel it later. If an action using the same key is already running, SceneKit removes it before adding the new action.
SceneKit calls your method after the action,Äôs duration is complete. For example, you can use this method with a wait action to execute some code after a timed delay. If during the delay period you need to prevent the code from running, use the removeActionForKey method to cancel it.
See also:

- 13.31.35 runAction(action as SCNActionMBS) 1026
- 13.31.36 runAction(action as SCNActionMBS, del as SCNNodeRunActionCompletedMBS, tag as variant = nil) 1027
- 13.31.37 runAction(action as SCNActionMBS, key as string) 1027

13.31.39 SCNModelTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix for transforming coordinates from model space to scene (or world) space.

13.31.40 SCNModelViewProjectionTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix containing the concatenation of the Model, View, and Projection transformations.

13.31.41 SCNModelViewTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix containing the concatenation of the Model and View transformations.

13.31.42 SCNNormalTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix for transforming surface normal vectors from model space to view (or eye) space.

13.31.43 SCNProjectionTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix for transforming coordinates from view (or eye) space to clip space.

13.31.44 SCNViewTransform as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the keys are used for the 'semantic' argument of SCNProgramMBS.setSemantic method.
Notes: Transforms are SCNMatrix4 wrapped in NSValue objects.
A 4 x 4 matrix for transforming coordinates from scene (or world) space to view (or eye) space.

13.31.45 setConstraints(constraints() as SCNConstraintMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the list of constraints affecting the node,Äôs transformation.
Notes: An array of constraint objects. Before rendering, SceneKit evaluates all constraints attached to a node hierarchy and adjusts node transformations appropriately.
Use the SCNLookAtConstraintMBS class to make a node always point toward another node even as both are moved, or the SCNTransformConstraintMBS class to apply arbitrary transformations at constraint eval-

uation time.

13.31.46 Properties

13.31.47 Camera as SCNCameraMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The camera attached to the node.
Notes: To use a camera for displaying a scene, set the the pointOfView property of the view (or layer or renderer) displaying the scene to the node containing the camera. A camera looks in the direction of the node,Äôs negative z-axis, so you aim the camera by changing the position and orientation of the node containing it. You control geometric and optical parameters of the camera—projection, field of view, and depth of field—using the attached SCNCameraMBS object.
(Read and Write property)

13.31.48 CastsShadow as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit renders the node,Äôs contents into shadow maps.
Notes: SceneKit renders shadows by rendering a shadow map image containing silhouettes of the scene,Äôs contents, and then projecting that image onto the scene. SceneKit performs this process once for each SCNLight object in the scene whose castsShadow property is true. Because shadow map rendering re-renders portions of the scene, it incurs a performance cost. To minimize this performance cost, exclude nodes from shadow map rendering by setting the node,Äôs castsShadow property to false.

For more details on shadow rendering, see SCNLight.
(Read and Write property)

13.31.49 categoryBitMask as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mask that defines which categories the node belongs to.
Notes: You can assign each node in a scene to one or more categories, where each category corresponds to a bit in the bit mask. You define the mask values used in your app. When SceneKit renders a scene, it compares the each node,Äôs categoryBitMask property with the category bit masks of every other object that participates in the rendering process—lights, cameras, and techniques—using a bitwise AND operation. If the result is a nonzero value, SceneKit includes the node when rendering. The default category bit mask is 1.

Use a node,Äôs category bit mask together with:

- An SCNLightMBS object,Äôs categoryBitMask property to exclude the node from that light,Äôs illumination
- An SCNCameraMBS object,Äôs categoryBitMask property to make the node invisible to that camera
- The category bit masks in an SCNTechniqueMBS object,Äôs definition dictionary to include or exclude the node from phases of a multipass rendering technique

This property does not affect SceneKit,Äôs physics simulation. To include or exclude a node from physics interactions, use the categoryBitMask property of the node,Äôs physicsBody and physicsField objects.
(Read and Write property)

13.31.50 EulerAngles as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node,Äôs orientation, expressed as pitch, yaw, and roll angles in radians. Animatable.
Notes: The order of components in this vector matches the axes of rotation:

- Pitch (the x component) is the rotation about the node,Äôs x-axis.
- Yaw (the y component) is the rotation about the node,Äôs y-axis.
- Roll (the z component) is the rotation about the node,Äôs z-axis.

SceneKit applies these rotations relative to the node,Äôs pivot property in the reverse order of the components: first roll, then yaw, then pitch. The rotation, eulerAngles, and orientation properties all affect the rotational aspect of the node,Äôs transform property. Any change to one of these properties is reflected in the others.
You can animate changes to this property,Äôs value. See Animating SceneKit Content.
(Read and Write property)

13.31.51 focusBehavior as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The focus behavior for a node.
Notes: (Read and Write property)

13.31.52 Geometry as SCNGeometryMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The geometry attached to the node.
Notes: A node can have only one geometry attached to it. To combine geometries so they can be controlled or animated together, create a node with no geometry and add other nodes to it.
Animating the node,Äôs geometric properties can move, rotate, stretch and scale its geometry. For more advanced animations of a node,Äôs geometry, use its morpher and skinner objects.
(Read and Write property)

13.31.53 Handle as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read and Write property)

13.31.54 Hidden as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines the visibility of the node,Äôs contents. Animatable.
Notes: The default value of this property is false, specifying that SceneKit should render geometries and use lights attached to the node or its children. Change this property,Äôs value to true to exclude attached geometries and lights from rendering. (Cameras attached to the node or its children are not affected by this property.) Hiding a node also hides its child nodes recursively.

You can animate changes to this property,Äôs value. See Animating SceneKit Content. Hiding or showing a node in an animation results in a fade-in or fade-out effect.
(Read and Write property)

13.31.55 Light as SCNLightMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The light attached to the node..
Notes: A node can have only one light attached to it. To combine lights so they can be controlled or animated together, create a node with no light and add other nodes to it.
(Read and Write property)

13.31.56 localFront as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The unit vector SceneKit treats as "forward" in local space for all nodes.
Notes: The "forward" direction of a node affects cameras and directional lighting attached to a node, as well as relative orientation and movement effects such as lookAt, SCNLookAtConstraint, and SCNBillboardConstraint.
This vector is always (0, 0, -1) for all nodes, but you can use this class property when it's convenient to refer to directions symbolically.

Available in macOS 10.13 or later.
(Read only property)

13.31.57 localRight as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The direction SceneKit treats as "right" in local space for all nodes.
Notes: No SceneKit features depend directly on this direction's definition—it's simply a natural consequence of recognizing "forward" and "up" directions for use with cameras, directional lighting, and relative orientation operations.
This vector is always (1, 0, 0) for all nodes, but you can use this class property when it's convenient to refer to directions symbolically.

Available in macOS 10.13 or later.
(Read only property)

13.31.58 localUp as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The direction SceneKit treats as "up" in local space for all nodes.
Notes: The "up" direction of a node affects cameras attached to a node, as well as relative orientation and movement effects such as lookAt, SCNLookAtConstraint, and SCNBillboardConstraint.
This vector is always (0, 1, 0) for all nodes, but you can use this class property when it's convenient to refer to directions symbolically.

Available in macOS 10.13 or later.
(Read only property)

13.31.59 movabilityHint as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A value that indicates how SceneKit should handle the node when rendering movement-related effects.
Notes: This property controls how the node contributes to various motion-related effects during rendering. The default value is SCNMovabilityHintFixed. If the movability hint is SCNMovabilityHintMovable:

- The node,Äôs content is not affected by motion blur effects. (See the motionBlurIntensity camera property, which applies motion blur effects for camera movement but not subject movement.)
- The node does not contribute to environmental lighting of other objects. Environmental lighting is an aspect of physically based rendering (see SCNLightingModelPhysicallyBased), in which surfaces tend to pick up the color of other scene content around them. SceneKit performs environmental lighting by rendering light probes—small, omnidirectional snapshots of the scene,Äôs contents, as visible from at various points around the scene. Light probes are static, so nodes that are expected to move should not contribute to the lighting environment for other surfaces.

This value is merely a hint that communicates to SceneKit,Äôs rendering system about how you want to move content in your scene; it does not affect your ability to change the node,Äôs position or add animations or physics to the node.
(Read and Write property)

13.31.60 Name as String

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A name associated with the node.
Notes: You can provide a descriptive name for a node to make managing your scene graph easier. Nodes loaded from a scene file may have names assigned by an artist using a 3D authoring tool. Use the childNodeWithName or childNodesPassingTest method to retrieve a node from a scene graph by its name, or the SCNSceneSource class to examine nodes in a scene file without loading its scene graph.

The names of nodes and their attached objects are saved when you export a scene to a file using its writeToURL method, and appear in the Xcode scene editor. The SceneKit statistics view (see showsStatistics) also shows the names of nodes with attached cameras.
(Read and Write property)

13.31.61 Opacity as Double

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The opacity value of the node. Animatable.
Notes: Possible values are between 0.0 (fully transparent) and 1.0 (fully opaque). The default is 1.0.
You can animate changes to this property,Äôs value.
(Read and Write property)

13.31.62 orientation as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node,Äôs orientation, expressed as a quaternion. Animatable.
Notes: The rotation, eulerAngles, and orientation properties all affect the rotational aspect of the node,Äôs transform property. Any change to one of these properties is reflected in the others.
You can animate changes to this property,Äôs value.
(Read and Write property)

13.31.63 parentNode as SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The node,Äôs parent in the scene graph hierarchy.
Notes: For a scene,Äôs rootNode object, the value of this property is nil.
(Read only property)

13.31.64 Paused as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether to run actions and animations attached to the node and its child nodes.
Notes: The default value of this property is false, specifying that SceneKit should continuously update the node,Äôs contents. Pausing a node pauses any running animations or actions. This property applies to the actions and animations attached to the node itself and those attached to any of its child or descendant nodes.
(Read and Write property)

13.31.65 physicsBody as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics body associated with the node.
Notes: The default value is nil, specifying that the node does not participate in the physics simulation at all. If you provide a physics body, SceneKit updates the node,Äôs position and orientation each time

it processes a step of its physics simulation. For more information on SceneKit,Äôs physics system, see SCNPhysicsWorldMBS.
(Read and Write property)

13.31.66 physicsField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics field associated with the node.
Notes: Physics fields apply forces to other physics bodies in a specified area around their node. For example, a SCNPhysicsDragField field slows all physics bodies that pass through its area. For a full list of field types and their effects, see SCNPhysicsFieldMBS.

A node can contain both a physics body that defines collision behavior and a physics field that defines forces in its area. For example, two nodes containing physics bodies and radial gravity fields will be attracted to one another, but will bounce off each other when they collide.
(Read and Write property)

13.31.67 pivot as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The pivot point for the node,Äôs position, rotation, and scale. Animatable.
Notes: A node,Äôs pivot is the transformation between its coordinate space and that used by its position, rotation, and scale properties. The default pivot is SCNMatrix4MBS.Identity, specifying that the node,Äôs position locates the origin of its coordinate system, its rotation is about an axis through its center, and its scale is also relative to that center point.
Changing the pivot transform alters these behaviors in many useful ways. You can:

- Offset the node,Äôs contents relative to its position. For example, by setting the pivot to a translation transform you can position a node containing a sphere geometry relative to where the sphere would rest on a floor instead of relative to its center.
- Move the node,Äôs axis of rotation. For example, with a translation transform you can cause a node to revolve around a faraway point instead of rotating around its center, and with a rotation transform you can tilt the axis of rotation.
- Adjust the center point and direction for scaling the node. For example, with a translation transform you can cause a node to grow or shrink relative to a corner instead of to its center.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.31.68 Position as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The translation applied to the node. Animatable.
Notes: The node,Äôs position locates it within the coordinate system of its parent, as modified by the node,Äôs pivot property. The default position is the zero vector, indicating that the node is placed at the origin of the parent node,Äôs coordinate system.
(Read and Write property)

13.31.69 presentationNode as SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A node object representing the state of the node as it currently appears onscreen.
Notes: When you use implicit animation (see SCNTransaction) to change a node,Äôs properties, those node properties are set immediately to their target values, even though the animated node content appears to transition from the old property values to the new. During the animation SceneKit maintains a copy of the node, called the presentation node, whose properties reflect the transitory values determined by any in-flight animations currently affecting the node. The presentation node,Äôs properties provide a close approximation to the version of the node that is currently displayed. SceneKit also uses the presentation node when computing the results of explicit animations, physics, and constraints.

Do not modify the properties of the presentation node. (Attempting to do so results in undefined behavior.) Instead, you use the presentation node to read current animation values—for example, to create a new animation starting at those values.

The presentation node has no parent or child nodes. To access animated properties of related nodes, use the node,Äôs own parentNode and childNodes properties and the presentationNode property of each related node.
(Read only property)

13.31.70 RenderingOrder as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The order the node,Äôs content is drawn in relative to that of other nodes.
Notes: Nodes with greater rendering orders are rendered last. Defaults to zero.
(Read and Write property)

13.31.71 Rotation as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node,Äôs orientation, expressed as a rotation angle about an axis. Animatable.
Notes: The four-component rotation vector specifies the direction of the rotation axis in the first three components and the angle of rotation (in radians) in the fourth. The default rotation is the zero vector, specifying no rotation. Rotation is applied relative to the node,Äôs pivot property.
The rotation, eulerAngles, and orientation properties all affect the rotational aspect of the node,Äôs transform property. Any change to one of these properties is reflected in the others.
(Read and Write property)

13.31.72 Scale as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The scale factor applied to the node. Animatable.
Notes: Each component of the scale vector multiplies the corresponding dimension of the node,Äôs geometry. The default scale is 1.0 in all three dimensions. For example, applying a scale of (2.0, 0.5, 2.0) to a node containing a cube geometry reduces its height and increases its width and depth. Scaling is applied relative to the node,Äôs pivot property.
(Read and Write property)

13.31.73 transform as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The transform applied to the node relative to its parent. Animatable.
Notes: The transformation is the combination of the node,Äôs rotation, position, and scale properties. The default transformation is SCNMatrix4MBS.Identity.

When you set the value of this property, the node,Äôs rotation, orientation, eulerAngles, position, and scale properties automatically change to match the new transform, and vice versa. SceneKit can perform this conversion only if the transform you provide is a combination of rotation, translation, and scale operations. If you set the value of this property to a skew transformation or to a nonaffine transformation, the values of these properties become undefined. Setting a new value for any of these properties causes SceneKit to compute a new transformation, discarding any skew or nonaffine operations in the original transformation.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.31.74 WorldFront as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The "forward" (-Z) direction vector relative to the node, expressed in world space.
Notes: Reading this property is equivalent to reading the localFront class property and using the convertVectorToNode or convertVectorFromNode method to convert that vector from the node's local coordinate space to the scene's world coordinate space.

Available in macOS 10.13 or later.
(Read only property)

13.31.75 worldOrientation as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node's orientation relative to the scene's world coordinate space.
Notes: This quaternion isolates the rotational aspect of the node's worldTransform matrix, which in turn is the conversion of the node's transform from local space to the scene's world coordinate space. That is, it expresses the difference in axis and angle of rotation between the node and the scene's rootNode.
(Read and Write property)

13.31.76 WorldPosition as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node's position relative to the scene's world coordinate space.
Notes: his vector isolates the translational aspect of the node's worldTransform matrix, which in turn is the conversion of the node's transform from local space to the scene's world coordinate space. That is, it expresses the x, y, and z offsets of the node's position from that of the scene's rootNode, and is equivalent to reading the node's position vector and converting it to world space with the convertPositionFromNode or convertPositionToNode method.
(Read and Write property)

13.31.77 WorldRight as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The "right" (+X) direction vector relative to the node, expressed in world space.
Notes: Reading this property is equivalent to reading the localRight class property and using the convertVectorToNode or convertVectorFromNode method to convert that vector from the node's local coordinate space to the scene's world coordinate space.

Available in macOS 10.13 or later.
(Read only property)

13.31.78 worldTransform as SCNMatrix4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The world transform applied to the node.
Notes: A world transform is the node,Äôs coordinate space transform relative to the scene,Äôs coordinate space. This transform is the concatenation of the node,Äôs transform property with that of its parent node, the parent,Äôs parent, and so on up to the rootNode object of the scene.
(Read and Write property)

13.31.79 WorldUp as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The ”up” (+Y) direction vector relative to the node, expressed in world space.
Notes: Reading this property is equivalent to reading the localUp class property and using the convertVectorToNode or convertVectorFromNode method to convert that vector from the node’s local coordinate space to the scene’s world coordinate space.

Available in macOS 10.13 or later.
(Read only property)

13.31.80 Constants

Movability Hint

Constant	Value	Description
SCNMovabilityHintFixed	0	The node is not expected to move over time.
SCNMovabilityHintMovable	1	The node is expected to move over time.

Focus Behavior Options

Constant	Value	Description
SCNNodeFocusBehaviorFocusable	2	Node is focusable and prevents nodes that it visually obscures from becomi focusable.
SCNNodeFocusBehaviorNone	0	Node is not focusable.
SCNNodeFocusBehaviorOccluding	1	Node is not focusable and prevents nodes that it visually obscures from becor ing focusable.

13.31.81 Delegates

13.31.82 SCNNodeRunActionCompletedMBS(node as SCNNodeMBS, action as SCNActionMBS, key as string, tag as variant)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The delegate for the runAction method.

13.32 class SCNPhysicsBallSocketJointMBS

13.32.1 class SCNPhysicsBallSocketJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A physics behavior that connects two physics bodies and allows them to pivot around each other in any direction.
Notes: A ball and socket joint has three rotational degrees of freedom and zero translational degrees of freedom. You can also use a ball and socket joint to pin a body to a specific location in the coordinate space of the node containing it while allowing it to rotate freely.

Available in MacOS 10.10.
Subclass of the SCNPhysicsBehaviorMBS class.

13.32.2 Methods

13.32.3 Constructor(body as SCNPhysicsBodyMBS, anchor as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a ball and socket joint that anchors a single physics body in space and allows it to rotate freely around an anchor point.
Notes: body: The physics body to be controlled by the joint.
anchor: The point the body pivots around, relative to the node containing it.

Returns a new ball-and-socket-joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.32.4 Constructor(bodyA as SCNPhysicsBodyMBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, anchorB as SCNVector3MBS) 1042

13.32.4 Constructor(bodyA as SCNPhysicsBodyMBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, anchorB as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a ball and socket joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.

anchorA: The point at which the joint connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.
anchorB: The point at which the joint connects, relative to the node containing the second body.

Returns a new ball-and-socket-joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior: method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.32.3 Constructor(body as SCNPhysicsBodyMBS, anchor as SCNVector3MBS) 1042

13.32.5 jointWithBody(body as SCNPhysicsBodyMBS, anchor as SCNVector3MBS) as SCNPhysicsBallSocketJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a ball and socket joint that anchors a single physics body in space and allows it to rotate freely around an anchor point.
Notes: body: The physics body to be controlled by the joint.
anchor: The point the body pivots around, relative to the node containing it.

Returns a new ball-and-socket-joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.32.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, anchorB as SCNVector3MBS) as SCNPhysicsBallSocketJointMBS 1043

13.32.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, anchorB as SCNVector3MBS) as SCNPhysicsBallSocketJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a ball and socket joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.
anchorA: The point at which the joint connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.

anchorB: The point at which the joint connects, relative to the node containing the second body.

Returns a new ball-and-socket-joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior: method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.32.5 jointWithBody(body as SCNPhysicsBodyMBS, anchor as SCNVector3MBS) as SCNPhysicsBallSocketJointMBS 1043

13.32.7 Properties

13.32.8 anchorA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the joint connects, relative to the node containing the first body.
Notes: (Read and Write property)

13.32.9 anchorB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the joint connects, relative to the node containing the second body.
Notes: (Read and Write property)

13.32.10 bodyA as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The first physics body connected by the joint.
Notes: (Read only property)

13.32.11 bodyB as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The second physics body connected by the joint.
Notes: This property,Äôs value is nil if the joint was created using the Constructor without bodyB param-

eter.
(Read only property)

13.33 class SCNPhysicsBehaviorMBS

13.33.1 class SCNPhysicsBehaviorMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The abstract superclass for joints, vehicle simulations, and other high-level behaviors that incorporate multiple physics bodies.
Notes: An SCNPhysicsBehavior object defines a high-level behavior for one or more physics bodies, modifying the results of the physics simulation. Behaviors include joints that connect multiple bodies so they move together and vehicle definitions that cause a body to roll like a car. You never use this class directly; instead, you instantiate one of the subclasses that defines the kind of behavior you want to add to your physics world.Table 1 describes the kinds of behaviors you can create in SceneKit.

Behavior classes implemented by SceneKit

Class Name	Description
SCNPhysicsHingeJointMBS	Connects two bodies and allows them to pivot around each other on a single axis.
SCNPhysicsBallSocketJointMBS	Connects two bodies and allows them to pivot around each other in any direction.
SCNPhysicsSliderJointMBS	Connects two bodies and allows them to slide or rotate relative to one another. Slider joints can also work as motors, applying a force or torque between the two bodies.
SCNPhysicsVehicleMBS	Simulates a physics body as the chassis of a car or other wheeled vehicle. You control a vehicle in terms of steering, braking, and acceleration, and use SCNPhysicsVehicleWheelMBS objects to define the appearance and physical properties of each of its wheels.

To use a physics behavior, you follow these steps:
1. Create SCNPhysicsBodyMBS objects and attach them to each node that participates in the behavior.
2. Create and configure a behavior object using one of the subclasses listed in above.
3. Add the behavior to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

13.33.2 Methods

13.33.3 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.33.4 Properties

13.33.5 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.34 class SCNPhysicsBodyMBS

13.34.1 class SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics simulation attributes attached to a scene graph node.
Notes: Overview
When SceneKit prepares to render a new frame, it performs physics calculations on physics bodies attached to nodes in the scene. These calculations include gravity, friction, and collisions with other bodies. You can also apply your own forces and impulses to a body. After SceneKit completes these calculations, it updates the positions and orientations of the node objects before rendering the frame.

To add physics to a node, create and configure an SCNPhysicsBody object and then assign it to the physicsBody property of the SCNNode object. A physics body must be associated with a node object before you apply forces or impulses to it.

A Body,Äôs Physical Characteristics
The SCNPhysicsBody class defines the physical characteristics for the body when it is simulated by the scene. Three properties are most important for physics simulation:

- The type property, which determines how the body interacts with forces and other bodies in the simulation. Static bodies are unaffected by forces and collisions and cannot move. Dynamic bodies are affected by forces and collisions with other body types. Kinematic bodies are not affected by forces or collisions, but by moving them directly you can cause collisions that affect dynamic bodies.
- The physicsShape property, which defines the three-dimensional form of the body for collision detection purposes. Physics simulations run faster when using simple shapes instead of the fine detail of a node,Äôs visible geometry. Typically, you set a body,Äôs physics shape to a bounding box, sphere, or primitive shape that roughly matches its node,Äôs visible content. For details on creating physics shapes, see SCNPhysicsShape.
- The kinematicBody property. Applying a force or torque to a dynamic body results in an acceleration (or angular acceleration) proportional to its mass.

All values in SceneKit,Äôs physics simulation use the International System of Units (SI): The unit of mass is the kilogram; the units of force, impulse, and torque are the newton, newton-second, and newton-meter; and the unit of distance for node positions and sizes is the meter. Note that you need not attempt to provide realistic values for physical quantities—use whatever values produce the behavior or gameplay you,Äôre looking for.
For a dynamic body, you can control how the body is affected by forces or collisions. See Defining How Forces Affect a Physics Body.

Defining a Body,Äôs Category and Collisions
When you design a game that uses physics, you define the different categories of physics objects that appear

in the scene. You define different categories of physics bodies for the behaviors your want for your app. A body can be assigned to as many of these categories as you want. In addition to declaring its own categories, a physics body also declares which categories of bodies it interacts with.
Use the categoryBitMask and collisionBitMask properties to define an object,Äôs collision behavior. The constants listed in SCNPhysicsCollisionCategory provide default values for these properties. In addition, with the contactTestBitMask property you can define interactions where a pair of bodies generates contact messages (see the SCNPhysicsContactDelegate protocol) without the bodies being affected by the collision.

Related Physics Classes
Physics fields create forces that affect all bodies in an area, such as vortices and gravitational attraction. For details and a list of available field types, see SCNPhysicsField.
You can add higher-level behaviors that control interactions between multiple bodies, such as joints and wheeled vehicles. For details and a list of available behaviors, see SCNPhysicsBehavior.
A scene,Äôs physicsWorld property holds an SCNPhysicsWorld object that manages physics characteristics that affect the entire scene.

Physics and the Rendering Loop
SceneKit evaluates its physics simulation as part of the rendering loop described in SCNSceneRendererDelegate. On each pass through this loop, SceneKit determines the state of all nodes with attached physics bodies, and simulates the effects of physics on those bodies for one time step—for example, by updating the position or rotation of a body based on its velocity and angular velocity. After simulating physics, SceneKit applies the results of the physics simulation to the scene for display.

Because you can animate SceneKit content not only through physics, but also through actions and implicitly and explicitly defined animations, SceneKit applies the results of physics simulation not to the SCNNode objects in your scene, but to each node,Äôs presentationNode object that represents its currently displayed state. As such, changing properties of a node that are affected by physics requires special consideration.

If you change the transform value—or any of the other properties that are components of the transform, such as position and rotation—of a node affected by physics, SceneKit resets the physics simulation for that node. If you want to change only one component of the transform, while leaving the others at their physics-simulated values, copy the presentation node,Äôs transform before making changes.

see also
https://developer.apple.com/documentation/scenekit/scnphysicsbody?language=objc
Blog Entries

- Physics example for SceneKit in Xojo

13.34.2 Methods

13.34.3 applyForce(direction as SCNVector3MBS, atPosition as SCNVector3MBS, impulse as Boolean)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Applies a force or impulse to the body at a specific point.
Notes: direction: The direction and magnitude of the force (in newtons) or of the impulse (in newton-seconds).
position: The point on the body where the force or impulse should be applied, in the local coordinate system of the SCNNodeMBS object containing the physics body.
impulse: True to apply an instantaneous change in momentum; false to apply a force that affects the body at the end of the simulation step.

Applying a force or impulse to a body at a position other than its center of mass may impart both linear and angular acceleration, depending on how the body is situated in the physics world and the other forces acting upon it.
The impulse parameter determines how this method contributes to the physics simulation:

- If you specify true, SceneKit treats the direction parameter as an impulse, measured in newton-seconds, and accelerates the physics body immediately. Use this option to simulate instantaneous effects such as launching a projectile.

- If you specify false, SceneKit treats the direction parameter as a force, measured in newtons. At the end of each simulation step (by default, a step occurs once for each frame in the rendering loop), SceneKit sums all forces applied to the physics body during that step and accelerates the body according to the net effect of those forces. Use this option when you want to simulate continuous forces on the body by calling applyForce on each simulation step.

Note
The impulse parameter effectively changes the unit of magnitude. A value that results in a certain acceleration when applied continuously on each frame of the simulation results in much less acceleration if applied only during a single frame.

As with all physical quantities in SceneKit, you need not use realistic force measurements in your app—the effects of the physics simulation depend on the relative differences between forces, not on their absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.
See also:

- 13.34.4 applyForce(direction as SCNVector3MBS, impulse as Boolean) 1051

13.34.4 applyForce(direction as SCNVector3MBS, impulse as Boolean)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Applies a force or impulse to the body at its center of mass.
Notes: direction: The direction and magnitude of the force (in newtons) or of the impulse (in newton-seconds).
impulse: True to apply an instantaneous change in momentum; false to apply a force that affects the body at the end of the simulation step.

Applying a force or impulse to a body imparts a linear acceleration proportional to its mass.
The impulse parameter determines how this method contributes to the physics simulation:

- If you specify true, SceneKit treats the direction parameter as an impulse, measured in newton-seconds, and accelerates the physics body immediately. Use this option to simulate instantaneous effects such as launching a projectile.
- If you specify false, SceneKit treats the direction parameter as a force, measured in newtons. At the end of each simulation step (by default, a step occurs once for each frame in the rendering loop), SceneKit sums all forces applied to the physics body during that step and accelerates the body according to the net effect of those forces. Use this option when you want to simulate continuous forces on the body by calling applyForce:impulse: on each simulation step.

Note
The impulse parameter effectively changes the unit of magnitude. A value that results in a certain acceleration when applied continuously on each frame of the simulation results in much less acceleration if applied only during a single frame.

As with all physical quantities in SceneKit, you need not use realistic force measurements in your app—the effects of the physics simulation depend on the relative differences between forces, not on their absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.
See also:

- 13.34.3 applyForce(direction as SCNVector3MBS, atPosition as SCNVector3MBS, impulse as Boolean) 1050

13.34.5 applyTorque(torque as SCNVector4MBS, impulse as Boolean)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Applies a net torque or a change in angular momentum to the body.
Notes: torque: The direction and magnitude of the torque (in newton-meters) or of the change of angular momentum (in newton-meter-seconds), relative to the world coordinate space of the scene.
impulse: True to apply an instantaneous change in angular momentum; false to apply a torque that affects

the body at the end of the simulation step.

Applying a torque to a body changes its angular velocity by an amount related to its mass and shape, rotating it without affecting its linear acceleration. Each component of the torque vector relates to rotation about the corresponding axis in the local coordinate system of the SCNNode object containing the physics body. For example, applying a torque of { 0.0, 0.0, 1.0 } causes a node to spin counterclockwise around the world-space z-axis.
The impulse parameter determines how this method contributes to the physics simulation:

- If you specify true, SceneKit treats the direction parameter as an instantaneous change in angular momentum, measured in newton-meter-seconds.
- If you specify false, SceneKit treats the direction parameter as a torque, measured in newton-meters. At the end of each simulation step (by default, a step occurs once for each frame in the rendering loop), SceneKit sums all forces and torques applied to the physics body during that step and accelerates the body according to the net effect of those forces and torques. Use this option when you want to simulate gradual acceleration by calling applyTorque:impulse: on each simulation step.

Note
The impulse parameter effectively changes the unit of magnitude. A value that results in a certain acceleration when applied continuously on each frame of the simulation results in much less acceleration if applied only during a single frame.

As with all physical quantities in SceneKit, you need not use realistic force and torque measurements in your app—the effects of the physics simulation depend on the relative differences between forces, not on their absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.

13.34.6 body(type as Integer, shape as SCNPhysicsShapeMBS) as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics body with the specified type and shape.
Notes: type: A constant that determines how a body responds to forces and collisions. See SCNPhysicsBodyType.
shape: A physics shape defining the volume of the body for collision detection purposes.

Returns a new physics body object.

For the body to participate in collision detection or respond to forces, you must attach it to the physicsBody property of an SCNNode object in a scene.
If you pass nil for the shape parameter, SceneKit automatically creates a physics shape for the body when

you attach it to a node, based on that node,Äôs geometry property. To create a physics shape that,Äôs based on the geometries of a node and its hierarchy of children, or to control the level of detail in a physics shape, create the physics shape manually using an SCNPhysicsShape class method.

Note
For nodes containing custom geometry, the physics shape SceneKit automatically creates is a rough approximation of the geometry. This approximation, or convex hull, provides a compromise between accuracy and performance in collision detection. For the best collision detection performance, create an SCNPhysicsShape instance based on a basic geometry class (SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, or SCNCapsule).

13.34.7 clearAllForces

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels all continuous forces and torques acting on the physics body during the current simulation step.
Notes: When you pass false for the impulse parameter in the applyForce, applyForce:atPosition, or applyTorque method, SceneKit waits until the end of the current simulation step before applying its effect. At that time, SceneKit sums all forces and torques applied during that simulation step and changes the velocity or angular velocity of the body according to the net effect of those forces and torques.
Call clearAllForces to cancel any forces and torques previously applied during the current simulation step.

13.34.8 Constructor(type as Integer, shape as SCNPhysicsShapeMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics body with the specified type and shape.
Notes: type: A constant that determines how a body responds to forces and collisions. See SCNPhysicsBodyType.
shape: A physics shape defining the volume of the body for collision detection purposes.

Returns a new physics body object.

For the body to participate in collision detection or respond to forces, you must attach it to the physicsBody property of an SCNNode object in a scene.
If you pass nil for the shape parameter, SceneKit automatically creates a physics shape for the body when you attach it to a node, based on that node,Äôs geometry property. To create a physics shape that,Äôs based on the geometries of a node and its hierarchy of children, or to control the level of detail in a physics shape, create the physics shape manually using an SCNPhysicsShape class method.

Note

For nodes containing custom geometry, the physics shape SceneKit automatically creates is a rough approximation of the geometry. This approximation, or convex hull, provides a compromise between accuracy and performance in collision detection. For the best collision detection performance, create an SCNPhysicsShape instance based on a basic geometry class (SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, or SCNCapsule).

13.34.9 copy as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the body.

13.34.10 dynamicBody as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics body that can be affected by forces and collisions.
Notes: Use dynamic bodies for the elements of your scene that are moved by the physics simulation.
For the body to participate in collision detection or respond to forces, you must attach it to the physicsBody property of an SCNNode object in a scene.
SceneKit automatically creates a physics shape for the body when you attach it to a node, based on that node,Äôs geometry property. To create a physics shape that,Äôs based on the geometries of a node and its hierarchy of children, or to control the level of detail in a physics shape, create the physics shape manually using an SCNPhysicsShape class method.

Note
For nodes containing custom geometry, the physics shape SceneKit automatically creates is a rough approximation of the geometry. This approximation, or convex hull, provides a compromise between accuracy and performance in collision detection. For the best collision detection performance, create an SCNPhysicsShape instance based on a basic geometry class (SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, or SCNCapsule).

13.34.11 kinematicBody as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics body that is unaffected by forces or collisions but that can cause collisions affecting other bodies when moved.
Notes: Return a new physics body object.

Use kinematic bodies for scene elements that you want to control directly but whose movement manipulates other elements. For example, to allow the user to push objects around with a finger, you might create a

kinematic body and attach it to an invisible node that you move follow touch events. (In macOS, use the same technique to allow the user to move objects with the mouse pointer.)

For the body to participate in collision detection or respond to forces, you must attach it to the physicsBody property of an SCNNode object in a scene.
SceneKit automatically creates a physics shape for the body when you attach it to a node, based on that node,Äôs geometry property. To create a physics shape that,Äôs based on the geometries of a node and its hierarchy of children, or to control the level of detail in a physics shape, create the physics shape manually using an SCNPhysicsShape class method.
Note
For nodes containing custom geometry, the physics shape SceneKit automatically creates is a rough approximation of the geometry. This approximation, or convex hull, provides a compromise between accuracy and performance in collision detection. For the best collision detection performance, create an SCNPhysicsShape instance based on a basic geometry class (SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, or SCNCapsule).

13.34.12 resetTransform

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Updates the position and orientation of a body in the physics simulation to match that of the node to which the body is attached.
Notes: If you change the position or orientation of a node with an attached static or dynamic physics body, call this method afterward to ensure that the physics simulation incorporates the change. You need not call this method for kinematic bodies.
Note that dynamic and physics bodies are designed to be moved only by the physics simulation or not at all. You may use this method to move them regardless of this restriction, but at a cost to performance.

13.34.13 staticBody as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics body that is unaffected by forces or collisions and that cannot move.
Notes: Use static bodies to construct fixtures in your scene that other bodies need to collide with but that do not themselves move, such as floors, walls, and terrain.
For the body to participate in collision detection or respond to forces, you must attach it to the physicsBody property of an SCNNode object in a scene.
SceneKit automatically creates a physics shape for the body when you attach it to a node, based on that node,Äôs geometry property. To create a physics shape that,Äôs based on the geometries of a node and its hierarchy of children, or to control the level of detail in a physics shape, create the physics shape manually using an SCNPhysicsShape class method.

Note

For nodes containing custom geometry, the physics shape SceneKit automatically creates is a rough approximation of the geometry. This approximation, or convex hull, provides a compromise between accuracy and performance in collision detection. For the best collision detection performance, create an SCNPhysicsShape instance based on a basic geometry class (SCNBox, SCNSphere, SCNPyramid, SCNCone, SCNCylinder, or SCNCapsule).

13.34.14 Properties

13.34.15 AffectedByGravity as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the constant gravity of a scene accelerates the body.
Notes: If this property is true (the default), and the type of the body is SCNPhysicsBodyTypeDynamic, the gravity property of the scene,Äôs physicsWorld object causes the body to accelerate.
If this property is false, the body is not affected by scene gravity. This option can be useful when making physics bodies whose behavior should be governed by SCNPhysicsFieldMBS objects instead of a constant global acceleration.
(Read and Write property)

13.34.16 allowsResting as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that specifies whether SceneKit can automatically mark the physics body at rest.
Notes: If true (the default), SceneKit keeps track of whether the body is moving or affected by forces, automatically setting its Resting property to true when it is ,Äúat rest.,Äù The physics simulation runs faster when simulating fewer bodies, so treating a body as resting temporarily removes it from the simulation to improve performance.

SceneKit automatically returns a resting body to the simulation if another body collides with it, if you change its position or velocity, or if you apply a force to it. However, SceneKit uses a faster, less accurate simulation when deciding whether to change a body,Äôs isResting property back to false. If testing your app reveals unexpected physics behaviors involving resting bodies, changing those bodies,Äô allowsResting property to false may improve simulation accuracy.
(Read and Write property)

13.34.17 angularDamping as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A factor that reduces the body,Äôs angular velocity.
Notes: This property simulates the effect of rotational friction on a body. A damping factor of 0.0 specifies no loss in angular velocity, and a damping factor of 1.0 prevents the body from rotating. The default damping factor is 0.1.
(Read and Write property)

13.34.18 angularVelocity as SCNVector4MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A vector describing both the current rotation axis and rotational speed (in radians per second) of the physics body.
Notes: A positive rotational speed indicates counterclockwise rotation (when viewed from the direction the rotation axis points in).
SceneKit,Äôs physics simulation determines the angular velocity (and corresponding change in rotation) of each dynamic physics body in the scene. You can also set a body,Äôs velocity directly to set the physics simulation in motion or influence its behavior.

The effect of reading or setting this property,Äôs value changes based on the current context:

- When invoked within a rendering loop method (any of the events), or from any other code invoked from within such a method, reading the property returns the current result of the physics simulation, and setting the property immediately applies the change.

- When invoked at any other time, reading the property returns the last value set for the property, and setting the property does not take effect until the next pass through the rendering loop.

(Read and Write property)

13.34.19 angularVelocityFactor as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A multiplier affecting how SceneKit applies rotations computed by the physics simulation to the node containing the physics body.
Notes: Use this property to constrain or restrict the effect of physics simulation on the node containing the physics body. For example, you can force a body to rotate in only one axis by setting its angular velocity factor to { 0.0, 1.0, 0.0 } .
(Read and Write property)

13.34.20 categoryBitMask as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mask that defines which categories this physics body belongs to.
Notes: Every physics body in a scene can be assigned to one or more categories, each corresponding to a bit in the bit mask. You define the mask values used in your game. Use this property together with the physicsShape and contactTestBitMask properties to define which physics bodies interact with each other and when your game is notified of interactions.
The default value is SCNPhysicsCollisionCategoryStatic for static bodies and SCNPhysicsCollisionCategoryDefault for dynamic and kinematic bodies.
(Read and Write property)

13.34.21 charge as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The electric charge of the body, in coulombs.
Notes: A body,Äôs charge determines its behavior when affected by an electric or magnetic field. Use the SCNPhysicsFieldMBS class to add these fields to your scene. Bodies with positive or negative charges behave differently when affected by electric or magnetic fields. The default electric charge on a physics body is 0.0, causing it to be unaffected by electric and magnetic fields.

Note that you need not use realistic measurements for the bodies in your app—the effects of the physics simulation depend on the relative masses of different bodies, not the absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.
(Read and Write property)

13.34.22 collisionBitMask as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mask that defines which categories of physics bodies can collide with this physics body.
Notes: When two physics bodies contact each other, a collision may occur. SceneKit compares the body,Äôs collision mask to the other body,Äôs category mask by performing a bitwise AND operation. If the result is a nonzero value, then the body is affected by the collision. Each body independently chooses whether it wants to be affected by the other body. For example, you might choose to avoid collision calculations that would make negligible changes to a body,Äôs velocity.

The default value is SCNPhysicsCollisionCategoryAll (a bit mask whose every bit is enabled), specifying that the body will collide with bodies of all other categories.
(Read and Write property)

13.34.23 contactTestBitMask as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mask that defines which categories of bodies cause intersection notifications with this physics body.
Notes: When two physics bodies overlap, a contact may occur. SceneKit compares the body,Äôs contact mask to the other body,Äôs category mask by performing a bitwise AND operation. If the result is a nonzero value, SceneKit creates an SCNPhysicsContactMBS object describing the contact and sends messages to the contactDelegate object of the scene,Äôs physics world. For best performance, only set bits in the contact mask for interactions you are interested in.

For applications running in OS X v10.10 or iOS 8, this property,Äôs value matches that of the collisionBitMask property—that is, SceneKit sends contact messages if and only if a collision occurs. For applications running in OS X v10.11 or iOS 9 or later, this property,Äôs value defaults to zero and need not match the collision mask—that is, a pair of bodies generates contact messages whenever the bodies intersect, regardless of whether they collide or pass through one another.
(Read and Write property)

13.34.24 damping as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A factor that reduces the body,Äôs linear velocity.
Notes: This property simulates the effect of fluid friction or air resistance on a body. A damping factor of 0.0 specifies no loss in velocity, and a damping factor of 1.0 prevents the body from moving. The default damping factor is 0.1.
(Read and Write property)

13.34.25 friction as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The body,Äôs resistance to sliding motion.
Notes: This property simulates the roughness of the body,Äôs surface. When two bodies are in contact and a force is applied that would cause them to slide against one another, the friction values for both bodies determine their resistance to motion. If both bodies,Äô friction value is 0.0, they slide freely against each other. If both bodies,Äô friction value is 1.0, they do not slide at all. The default friction is 0.5.
(Read and Write property)

13.34.26 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.34.27 isResting as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the physics body is at rest.
Notes: This property,Äôs default value is false, but SceneKit,Äôs physics simulation may automatically set it to true if the body is not moving and not affected by any forces. A resting body does not participate in the simulation until another body collides with it or you change its position or velocity or apply a force to it.
(Read only property)

13.34.28 mass as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The mass of the body, in kilograms.
Notes: The mass of a body affects its momentum and how it responds to forces. The default mass for dynamic bodies is 1.0. The default mass for static and kinematic bodies is 0.0, but these bodies are unaffected by mass.

Note that you need not use realistic measurements for the bodies in your app—the effects of the physics simulation depend on the relative masses of different bodies, not the absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.
(Read and Write property)

13.34.29 momentOfInertia as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The body,Äôs moment of inertia, expressed in the local coordinate system of the node that contains the body.
Notes: A body,Äôs moment of inertia determines how it responds to torques (that is, forces with a rotational component). Each component of this vector is the moment of inertia for the corresponding principal axis (in the coordinate system containing the physics body). For example, if the x-component value of the moment vector is less than the y-component value, the body rotates more freely about its x-axis than its

y-axis.

By default, SceneKit automatically determines the body,Äôs moment of inertia based on its shape and mass. Use this property to define a custom moment of inertia (for example, to model an object of non-uniform density). Using a custom moment of inertia requires setting the usesDefaultMomentOfInertia property to false.
(Read and Write property)

13.34.30 physicsShape as SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that defines the solid volume of the physics body for use in collision detection.
Notes: The physics simulation does not use a node,Äôs visible geometry for collision detection—the simulation can run faster when using simple shapes, and it can also be useful to design your app or game using invisible collision shapes for some elements. Typically, you set a body,Äôs physics shape to a bounding box or primitive shape that roughly matches its node,Äôs visible content, but you can use a more detailed shape for more precise collision detection at a cost to performance.
For details on creating physics shapes, see SCNPhysicsShapeMBS.
(Read and Write property)

13.34.31 restitution as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A factor that determines how much kinetic energy the body loses or gains in collisions.
Notes: This property simulates the ,Äúbounciness,Äù of a body. A restitution of 1.0 means that the body loses no energy in a collision—for example, a ball dropped onto a flat surface will bounce back to the height it fell from. A restitution of 0.0 means the body does not bounce after a collision. A restitution of greater than 1.0 causes the body to gain energy in collisions. The default restitution is 0.5.
(Read and Write property)

13.34.32 rollingFriction as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The body,Äôs resistance to rolling motion.
Notes: This property simulates the traction between a rounded body and bodies it might roll against. A rolling friction of 0.0 (the default) means that a body induced to roll (for example, by being placed on an inclined surface) will continue to roll without slowing down unless otherwise acted upon, and a rolling friction of 1.0 prevents the body from rolling.
(Read and Write property)

13.34.33 type as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A constant that determines how the physics body responds to forces and collisions.
Notes: (Read and Write property)

13.34.34 usesDefaultMomentOfInertia as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit automatically calculates the body,Äôs moment of inertia or allows setting a custom value.
Notes: A body,Äôs moment of inertia determines how it responds to torques (that is, forces with a rotational component).
If this property is true (the default), SceneKit automatically determines the body,Äôs moment of inertia based on its shape and mass. Set this property to false and use the momentOfInertia property to define a custom moment of inertia (for example, to model an object of non-uniform density).
(Read and Write property)

13.34.35 velocity as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A vector describing both the current speed (in meters per second) and direction of motion of the physics body.
Notes: SceneKit,Äôs physics simulation determines the velocity (and corresponding change in position) of each dynamic physics body in the scene. You can also set a body,Äôs velocity directly to set the physics simulation in motion or influence its behavior.
The effect of reading or setting this property,Äôs value changes based on the current context:

- When invoked within a rendering loop method (any of the events), or from any other code invoked from within such a method, reading the property returns the current result of the physics simulation, and setting the property immediately applies the change.
- When invoked at any other time, reading the property returns the last value set for the property, and setting the property does not take effect until the next pass through the rendering loop.

(Read and Write property)

13.34.36 velocityFactor as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A multiplier affecting how SceneKit applies translations computed by the physics simulation to the node containing the physics body.
Notes: Use this property to constrain or restrict the effect of physics simulation on the node containing the physics body. For example, you can force a body to move in only two dimensions by setting its velocity factor to { 1.0, 1.0, 0.0 } .
(Read and Write property)

13.34.37 Constants

Collision Categories

Constant	Value	Description
kPhysicsCollisionCategoryAll	-1	This is the default value for a physics body,Äôs collisionBitMask property. With this collision mask, a physics body can collide with all other physic bodies.
kPhysicsCollisionCategoryDefault	1	The default categoryBitMask value for dynamic and kinematic bodies.
kPhysicsCollisionCategoryStatic	2	The default categoryBitMask value for static bodies.

Body Types

Constant	Value	Description
kTypeDynamic	1	A physics body that can be affected by forces and collisions. Use dynamic bodies for the elements of your scene that are moved by the physics simulation.
kTypeKinematic	2	A physics body that is unaffected by forces or collisions but that can cause collisions affecting other bodies when moved. Use kinematic bodies for scene elements that you want to control directly directly but whose movement manipulates other elements. For example, to allow the user to push objects around with a finger, you might create a kinematic body and attach it to an invisible node that you move to follow touch events. (In macOS, use the same technique to allow the user to move objects with the mouse pointer.)
kTypeStatic	0	A physics body that is unaffected by forces or collisions and cannot move. Use static bodies to construct fixtures in your scene that other bodies need to collide with but that do not themselves move, such as floors, walls, and terrain.

13.35 class SCNPhysicsConeTwistJointMBS

13.35.1 class SCNPhysicsConeTwistJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The class for a cone twist join.
Notes: Subclass of the SCNPhysicsBehaviorMBS class.

13.35.2 Methods

13.35.3 Constructor(body as SCNPhysicsBodyMBS, frame as SCNMatrix4MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a physics cone-twist joint.
Notes: The joint attaches "body" to the 3d location specified by "frame" and relative to the node that owns the body.
See also:

- 13.35.4 Constructor(bodyA as SCNPhysicsBodyMBS, frameA as SCNMatrix4MBS, bodyB as SCNPhysicsBodyMBS, frameB as SCNMatrix4MBS) 1064

13.35.4 Constructor(bodyA as SCNPhysicsBodyMBS, frameA as SCNMatrix4MBS, bodyB as SCNPhysicsBodyMBS, frameB as SCNMatrix4MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a physics cone-twist joint.
Notes: The joint attaches bodyA and bodyB on frameA and frameB respectively.
See also:

- 13.35.3 Constructor(body as SCNPhysicsBodyMBS, frame as SCNMatrix4MBS) 1064

13.35.5 jointWithBody(body as SCNPhysicsBodyMBS, frame as SCNMatrix4MBS) as SCNPhysicsConeTwistJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a physics cone-twist joint.
Notes: The joint attaches "body" to the 3d location specified by "frame" and relative to the node that owns the body.
See also:

13.35.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, frameA as SCNMatrix4MBS, bodyB as SCNPhysicsBodyMBS, frameB as SCNMatrix4MBS) as SCNPhysicsConeTwistJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Initializes and returns a physics cone-twist joint.
Notes: The joint attaches bodyA and bodyB on frameA and frameB respectively.
See also:

13.35.7 Properties

13.35.8 bodyA as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The first body attached to the slider joint.
Notes: (Read only property)

13.35.9 bodyB as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The second body attached to the slider joint.
Notes: (Read only property)

13.35.10 frameA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Frame for bodyA.
Notes: (Read and Write property)

13.35.11 frameB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Frame for bodyB.
Notes: (Read and Write property)

13.35.12 maximumAngularLimit1 as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum angular limits in radians in each cone tangent directions
Notes: (Read and Write property)

13.35.13 maximumAngularLimit2 as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum angular limits in radians in each cone tangent directions
Notes: (Read and Write property)

13.35.14 maximumTwistAngle as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Maximum twist angle alon the cone axis.
Notes: (Read and Write property)

13.36 class SCNPhysicsContactMBS

13.36.1 class SCNPhysicsContactMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Detailed information about a contact between two physics bodies in a scene,Äôs physics simulation.
Notes: You don't create SCNPhysicsContact instances directly; SceneKit automatically creates these objects whenever contacts occur.
To receive contact messages, implement events in your of your scene,Äôs SCNPhysicsWorld object. Next, for each physics body in your scene, set the categoryBitMask and collisionBitMask properties to define which interactions should generate contact messages.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

13.36.2 Methods

13.36.3 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.36.4 Properties

13.36.5 collisionImpulse as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The force over time of the collision, in newton-seconds.
Notes: This property,Äôs value tells you how hard the bodies struck each other in a collision. For example, in a game you might allow a character to proceed unhindered after a minor collision, but take damage when struck with sufficient force.
(Read only property)

13.36.6 contactNormal as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The normal vector at the contact point between the two physics bodies, in scene coordinates.
Notes: This vector tells you which direction the bodies were moving relative to one another at the time of the collision. For example, in a game you can examine this vector to have enemy characters take damage

when struck from above by the player character but damage the player character instead when they collide side-to-side.
(Read only property)

13.36.7 contactPoint as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The contact point between the two physics bodies, in scene coordinates.
Notes: (Read only property)

13.36.8 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.36.9 nodeA as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The node containing the first body in the contact.
Notes: Use the node,Äôs physicsBody property to examine physics characteristics of the node.
(Read only property)

13.36.10 nodeB as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The node containing the second body in the contact.
Notes: Use the node,Äôs physicsBody property to examine physics characteristics of the node.
(Read only property)

13.36.11 penetrationDistance as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The distance of overlap, in units of scene coordinate space, between the two physics bodies.
Notes: (Read only property)

13.36.12 sweepTestFraction as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Value between 0 and 1 giving the relative position of the physic shape when performing a convex sweep test.
Notes: (Read only property)

13.37 class SCNPhysicsFieldMBS

13.37.1 class SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An object that applies forces, such as gravitation, electromagnetism, and turbulence, to physics bodies within a certain area of effect.
Notes: You can create many types of field effects, such as gravitation, electromagnetism, and turbulence. To add a field effect to a scene, you create a physics field of the type you want to use and then attach it to the physicsField property of a node in the scene.
Physics fields can affect both SCNPhysicsBodyMBS objects and the particles spawned by SCNParticleSystemMBS objects.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

13.37.2 Methods

13.37.3 Constructor

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.37.4 copy as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the field.

13.37.5 customField(FieldForceEvaluator as SCNFieldForceEvaluatorMBS, tag as variant = nil) as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that runs the specified block to determine the force a field applies to each object in its area of effect.
Notes: delegate: A method that SceneKit runs for each object in the field,Äôs area of effect.

Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNodeMBS object.

For custom physics fields, SceneKit ignores the direction, strength, falloffExponent, and minimumDistance properties. Instead, SceneKit calls your block to determine the direction and magnitude of force to apply to each physics body or particle in the field,Äôs area of effect.

Due to the dispatching of the delegate to main thread, this is not very efficient.

13.37.6 dragField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that slows any object in its area of effect with a force proportional to the object,Äôs velocity.
Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Like the damping and angularDamping properties of a physics body, drag fields can simulate effects such as fluid friction or air resistance. Unlike those properties, drag fields can simulate different intensities of fluid friction in different areas of your scene. For example, you can use a drag field to represent underwater areas. The default falloffExponent value for a drag field is 0.0, indicating that the field,Äôs effect is constant throughout its area of effect.

13.37.7 electricField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that attracts or repels objects based on their electrical charge and on their distance from the field,Äôs center.
Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Use this field type to make objects behave differently from one another when they enter a region, or to make an object's behavior different from its mass-based behavior. An electric field behaves according to the first part of the Lorentz force equation modeling real-world electromagnetic forces—the field applies a force whose magnitude is proportional to electric charge and distance.
By default, physics bodies and particle systems have no electric charge, so they are unaffected by electric and magnetic fields. Use the charge property of a physics body or the particleCharge property of a particle system to add charge-based behavior.
When the field,Äôs strength value is positive (the default), it attracts bodies whose charge is negative and repels bodies whose charge is positive. To reverse this behavior, set the field,Äôs strength property to a negative value.
The default falloffExponent value for an electric field is 2.0, indicating that the field,Äôs effect diminishes with the square of its distance from its center.

13.37.8 linearGravityField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that accelerates objects in a specific direction.
Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Because the force of gravity on an object is proportional to the object,Äôs mass, this force accelerates all objects in the field,Äôs area of affect by the same amount. The field,Äôs strength property measures this acceleration in meters per second per second.
By default, a linear gravity field accelerates objects in along its direction vector. To make it accelerate objects in the opposite direction, set the field,Äôs strength property to a negative value.
The default falloffExponent value for a linear gravity field is 0.0, indicating that the field,Äôs effect is constant throughout its area of effect.

13.37.9 magneticField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that attracts or repels objects based on their electrical charge, velocity, and distance from the field,Äôs axis.
Notes: A physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Use this field type to make objects behave differently from one another when they enter a region, or to make an object's behavior different from its mass based behavior. A magnetic field behaves according to the second part of the Lorentz force equation modeling real-world electromagnetic forces—the field applies a force determined by the cross product of an object,Äôs velocity vector and the magnetic field vector at the object,Äôs location, with magnitude proportional to the object,Äôs electric charge.
By default, physics bodies and particle systems have no electric charge, so they are unaffected by electric and magnetic fields. Use the charge property of a physics body or the particleCharge property of a particle system to add charge-based behavior.
When the field,Äôs strength value is positive (the default), the magnetic field vectors circulate counterclockwise relative to the field,Äôs direction vector. (That is, the magnetic field models a real-world magnetic field created by current in a wire oriented in the field,Äôs direction.) To make field vectors circulate clockwise, set the field,Äôs strength property to a negative value.

This SCNPhysicsField option models the real-world physics effect of magnetic fields on moving, electrically charged bodies, not the behavior of permanent magnets or electromagnets. To make objects in your scene simply attract or repel one another, use a different field type. For example, a field created by the radialGravityField method attracts or repels all dynamic bodies near it according to its strength property, and a field created by the electricField method selectively attracts or repels bodies according to their electric charge.
The default falloffExponent value for a magnetic field is 2.0, indicating that the field,Äôs effect diminishes with the square of distance from its center.

13.37.10 noiseField(smoothness as double, animationSpeed as double) as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that applies random forces to objects in its area of effect.
Notes: smoothness: The amount of randomness in the field. A value of 0.0 specifies maximum noise, and a value of 1.0 specifies no noise at all.
speed: The field,Äôs variation over time. Specify 0.0 for a static field.

Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Use this field type to simulate effects involving random motion, such as fireflies or gently falling snow.
In calculating the direction and strength of the field,Äôs effect on an object, SceneKit uses a Perlin simplex noise function. This function produces a velocity field that varies over time.
The default falloffExponent value for a noise field is 0.0, indicating that the field,Äôs effect is constant throughout its area of effect. This field type ignores the field,Äôs direction property.

13.37.11 radialGravityField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that accelerates objects toward its center.
Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Because the force of gravity on an object is proportional to the object,Äôs mass, this force accelerates all objects at the same distance from the field,Äôs center by the same amount. The field,Äôs strength property measures this acceleration in meters per second per second.
By default, a radial gravity field attracts objects toward its center. To make it repel objects instead, set the field,Äôs strength property to a negative value.
The default falloffExponent value for a radial gravity field is 2.0, indicating that the field,Äôs effect diminishes with the square of distance from its center.

13.37.12 springField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that pulls objects toward its center with a spring-like force.

Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

The force a spring field applies to objects in its area of effect is linearly proportional to the distance from the object to the center of the field. (That is, the field behaves according to Hooke,Äôs Law of real-world spring forces.) An object placed at the center of the field and moved away will oscillate around the center, with a period of oscillation that is proportional to the object,Äôs mass. The field,Äôs strength property scales the magnitude of the spring effect—a larger strength simulates a stiffer spring.

The default falloffExponent value for a spring field is 1.0, indicating that the field,Äôs effect diminishes linearly with distance from its center.

13.37.13 turbulenceField(smoothness as double, animationSpeed as double) as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field that applies random forces to objects in its area of effect, with magnitudes proportional to those objects,Äô velocities.
Notes: smoothness: The amount of randomness in the field. A value of 0.0 specifies maximum noise, and a value of 1.0 specifies no noise at all.
speed: The field,Äôs variation over time. Specify 0.0 for a static field.

Return a physics field object. To use the field in a scene, attach it to the physicsField property of an SCNNode object.

Like a noise field, a turbulence field applies forces in random directions to the objects that it affects. Unlike a noise field, a turbulence field applies a force whose magnitude is proportional to the speed of each affected object. For example, an object passing through a noise field shakes as it travels through the field, but an object passing through a turbulence field shakes more violently the faster it travels. The field,Äôs strength property scales the magnitude of the turbulence effect.

The default falloffExponent value for a turbulence field is 0.0, indicating that the field,Äôs effect is constant throughout its area of effect.

13.37.14 vortexField as SCNPhysicsFieldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a field whose forces circulate around an axis.
Notes: Returns a physics field object. To use the field in a scene, attach it to the physicsField property of

an SCNNode object.

The force on an object in a vortex field is tangential to the line from the object,Äôs position to the field,Äôs axis and proportional to the object,Äôs mass. (The field,Äôs axis is a line that is parallel to its direction vector and that passes through its center. For details, see the offset property.) For example, when a vortex field,Äôs area of effect contains many objects, the resulting scene resembles a tornado: The objects simultaneously revolve around and fly away from the field,Äôs center.
By default, a vortex circulates counterclockwise relative to its direction vector. To make it circulate clockwise, set the field,Äôs strength property to a negative value.
The default falloffExponent value for a vortex field is 0.0, indicating that the field,Äôs effect is constant throughout its area of effect.

13.37.15 Properties

13.37.16 Active as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the field,Äôs effect is enabled.
Notes: If this value is fasle, the field does not apply forces to physics bodies in its area of effect. The default value is true.
Use this property, for example, to switch fields on and off as a gameplay mechanic.
(Read and Write property)

13.37.17 CategoryBitMask as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A mask that defines which categories this physics field belongs to.
Notes: To determine whether a field affects a physics body, SceneKit performs a bitwise AND operation on the field,Äôs category bit mask and the body,Äôs categoryBitMask property. If the result is a nonzero value, SceneKit computes and applies the force of the field on the body. To determine whether a field affects the particles spawned by an SCNParticleSystemMBS object, SceneKit performs the same check using the categoryBitMask property of the node containing the particle system.
Use this property to create fields which affect only certain bodies in your scene. Reducing the number of bodies affected by fields can also improve simulation performance.
(Read and Write property)

13.37.18 direction as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The field,Äôs directional axis.
Notes: Some types of fields apply forces whose direction or magnitude is relative to an axis. (For details on each type of field, see the methods listed in Creating Physics Fields.) Changing the direction changes the effects of these field types.

The default direction is the vector { 0, -1, 0 } . With this direction, for example, linear gravity fields whose strength is positive cause objects to fall in the ‚Äúdown‚Äù direction of scene space.
(Read and Write property)

13.37.19 Exclusive as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the field overrides other fields whose areas of effect it overlaps.
Notes: If this value is true and a physics body is within this field,Äôs region, SceneKit ignores the effects of all other fields that might otherwise affect the body. The default value is false.
If you set this property to true on multiple fields in a scene, their regions should not overlap. If they do, the results are undefined.
(Read and Write property)

13.37.20 falloffExponent as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An exponent that determines how the field,Äôs strength diminishes with distance.
Notes: When SceneKit calculates the force to be applied by a field, it attenuates the field,Äôs effect by multiplying with the expression pow(distance - minRadius, -falloff). If the falloff exponent is greater than zero, the field,Äôs effect is stronger on nearby bodies than on bodies farther away from its location.
The default falloff exponent varies by field type. For details, see the methods listed in Creating Physics Fields.
(Read and Write property)

13.37.21 halfExtent as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A location marking the end of the field,Äôs area of effect.
Notes: To define a field,Äôs area of effect, use the position property of the node that contains the field and the field,Äôs halfExtent property. The center of the area of effect is the node,Äôs position. The half-extent, a position vector in the local coordinate space of the node containing the field, marks one corner of a box, and the negative of the half-extent vector marks the opposite corner of the box. For example, if a node,Äôs

position is the vector { 2.0, 2.0, 2.0 } and it contains a physics field whose half-extent is the vector { 0.5, 0.5, 0.5 } , the field,Äôs area of effect is the box extending from 1.5 to 2.5 along each axis of the scene,Äôs coordinate system.
By default, a field,Äôs area of effect is the interior of this box shape. Use the usesEllipsoidalExtent property to instead make the area of effect an ellipsoid bounded by this box. Use the scope property to choose whether the area of effect is the interior or exterior of the box (or ellipsoid).
The default half-extent is the vector { INFINITY, INFINITY, INFINITY } , specifying that the field affects bodies located anywhere in the scene.
(Read and Write property)

13.37.22 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.37.23 minimumDistance as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum value for distance-based effects.
Notes: This property determines the beginning of the field,Äôs falloff area. At distances less than the minimum, the field,Äôs effect is at full strength. At greater distances, the field,Äôs effect diminishes based on the value of the falloffExponent property. The default minimum distance is a very small (but nonzero) value.
(Read and Write property)

13.37.24 offset as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The offset of the field,Äôs center within its area of effect.
Notes: Some types of fields apply forces whose magnitude is relative to the distance between an object and the field,Äôs center. (For details on each type of field, see the methods listed in Creating Physics Fields.) Changing the offset changes the effects of these field types.

With the default offset vector { 0, 0, 0 } , the center of a field is the center of its area of effect.
(Read and Write property)

13.37.25 Scope as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The area affected by the field, either inside or outside its region.
Notes: First, define a field,Äôs region using its halfExtent property and the position property of the node containing the field. Then, use the scope property to choose whether the field,Äôs area of effect is the interior of the region (the default) or all space outside the region.
(Read and Write property)

13.37.26 strength as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A multiplier for the force that the field applies to objects in its area of effect.
Notes: Each type of physics field defines its own behavior for strength values. For details, see the methods listed in Creating Physics Fields.
(Read and Write property)

13.37.27 usesEllipsoidalExtent as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the field,Äôs area of effect is shaped like a box or ellipsoid.
Notes: If this value is false (the default), the field,Äôs area of effect is the box-shaped region of space defined by its halfExtent property and the position property of the node containing the field.

If this value is true, the field,Äôs area of effect is the ellipsoid bounded by this box-shaped region. That is, if all components of the half-extent vector are equal, the field has a spherical area of effect.
(Read and Write property)

13.37.28 Constants

Scopes

Constant	Value	Description
kScopeInsideExtent	0	The field,Äôs effect applies only to objects within the region of space defined by its position and extent.
kScopeOutsideExtent	1	The field,Äôs effect applies only to objects outside the region of space defined by its position and extent.

13.37.29 Delegates

13.37.30 SCNFieldForceEvaluatorMBS(position as SCNVector3MBS, velocity as SCNVector3MBS, mass as single, charge as single, time as double, Tag as Variant) as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The signature for a delegate that SceneKit calls to determine the effect of a custom field on an object.
Notes: You use this type of delegate to create a custom physics field with the customFieldWithEvaluation-Block: method. SceneKit calls your delegate once for each object in the field,Äôs area of effect, on each step of the physics simulation.

Note
By default, one simulation step occurs for each frame rendered. For example, if your view renders at 60 frames per second and three bodies are in the field,Äôs area of effect, SceneKit runs your delegate 180 times per second. To avoid reduced rendering performance, take care not to perform extensive computation in this delegate.
The delegate takes the following parameters:

position: The position of the object affected by the field, in the local coordinate space of the node containing the field.
velocity: The velocity of the object affected by the field, relative to the local coordinate space of the node containing the field.
mass: The mass of the object affected by the field. (See the mass property for physics bodies and the particleMass property for particle systems.)
charge: The electrical charge of the object affected by the field. (See the charge property for physics bodies and the particleCharge property for particle systems.)
time: The elapsed time, in seconds, since the last simulation step.

Your delegate uses these parameters to compute and return an SCNVector3 force vector, which SceneKit then applies to the object affected by the field.

13.38 class SCNPhysicsHingeJointMBS

13.38.1 class SCNPhysicsHingeJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A physics behavior that connects two bodies and allows them to pivot around each other on a single axis.
Notes: A hinge has a single degree of freedom (rotation). You can also use a hinge joint to pin a body so that it can only move by rotating around a specific axis in the coordinate space of the node containing it.

Available in MacOS 10.10 or newer.
Subclass of the SCNPhysicsBehaviorMBS class.

13.38.2 Methods

13.38.3 Constructor(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a hinge joint that anchors a single physics body in space and lets it rotate around a specific axis.
Notes: body: The physics body to be controlled by the hinge joint.
axis: The direction of the axis that the body pivots around, relative to the node containing the body.
anchor: The location of the axis in the node containing the body.

Returns a new hinge joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.38.4 Constructor(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) 1081

13.38.4 Constructor(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a hinge joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.
axisA: The axis that the hinge pivots around, relative to the node containing the first body.
anchorA: The point at which the hinge connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.
axisB: The axis that the hinge pivots around, relative to the node containing the second body.
anchorB: The point at which the hinge connects, relative to the node containing the second body.

Returns a new hinge joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.38.3 Constructor(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) 1080

13.38.5 jointWithBody(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) as SCNPhysicsHingeJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a hinge joint that anchors a single physics body in space and lets it rotate around a specific axis.
Notes: body: The physics body to be controlled by the hinge joint.
axis: The direction of the axis that the body pivots around, relative to the node containing the body.
anchor: The location of the axis in the node containing the body.

Returns a new hinge joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.38.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) as SCNPhysicsHingeJointMBS 1082

13.38.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) as SCNPhysicsHingeJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a hinge joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.
axisA: The axis that the hinge pivots around, relative to the node containing the first body.
anchorA: The point at which the hinge connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.
axisB: The axis that the hinge pivots around, relative to the node containing the second body.
anchorB: The point at which the hinge connects, relative to the node containing the second body.

Returns a new hinge joint behavior.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.38.5 jointWithBody(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) as SCNPhysicsHingeJointMBS 1081

13.38.7 Properties

13.38.8 anchorA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the hinge connects, relative to the node containing the first body.
Notes: (Read and Write property)

13.38.9 anchorB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the hinge connects, relative to the node containing the second body.
Notes: (Read and Write property)

13.38.10 axisA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The axis that the hinge pivots around, relative to the node containing the first body.
Notes: (Read and Write property)

13.38.11 axisB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The axis that the hinge pivots around, relative to the node containing the second body.
Notes: (Read and Write property)

13.38.12 bodyA as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The first physics body connected by the joint.
Notes: (Read only property)

13.38.13 bodyB as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The second physics body connected by the joint.
Notes: This property,Äôs value is nil if the joint was created using the Constructor without bodyB parameter.
(Read only property)

13.39 class SCNPhysicsShapeMBS

13.39.1 class SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An abstraction of a physics body,Äôs solid volume for tuning collision detection.
Notes: When SceneKit performs contact detection and other simulations for the SCNPhysicsBodyMBS objects in your scene, it uses physics shapes instead of the rendered geometry of visible objects. This approach both improves simulation performance and allows you to more easily design your gameplay around scene elements the player can interact with.

Simple Versus Complex Shapes
When you allow SceneKit to automatically create a physics shape, it uses the simplest possible shape roughly matching the geometry of the node the physics body is attached to. This approach maximizes simulation performance but can lead to unrealistic physics behavior for some objects.

You can make the simulation behave more realistically by defining physics shapes that more closely follow the visible geometry in your scene. This approach comes at a cost to performance, so you want to limit the amount of detail in your physics shapes. Use the highest levels of detail only on bodies for which precise collision detection is important for your app.
If you create a physics shape using one of the basic geometry classes (SCNBoxMBS, SCNSphereMBS, SCNPyramidMBS, SCNConeMBS, SCNCylinderMBS, or SCNCapsuleMBS), SceneKit uses an idealized form of that geometry for the physics shape instead of using the geometry,Äôs vertex data to simulate collisions. For example, if you create a physics shape from an SCNSphere object, SceneKit simulates collisions for any object that passes within the sphere,Äôs radius.

Because the idealized forms of simple geometries are computationally much simpler than the vertex data needed for displaying them, using basic geometries for physics shapes (or compound shapes created from basic geometries with the shapeWithShapes method) often provides the best balance between simulation accuracy and performance.

Changing a Physics Body,Äôs Shape
Physics shapes are immutable, but you can change the shape associated with a physics body by creating a new SCNPhysicsShape instance and assigning it to the body,Äôs physicsShape property.
Blog Entries

- Physics example for SceneKit in Xojo

13.39.2 Methods

13.39.3 Constructor(geometry as SCNGeometryMBS, Options as Dictionary = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics shape based on a geometry object.
Notes: geometry: A geometry object.
options: A dictionary of options affecting the level of detail of the physics shape, or nil to use default options. For applicable keys and their possible values, see Shape Creation Options Keys.

Returns a new physics shape object.

If you create a physics shape using one of the basic geometry classes (SCNBoxMBS, SCNSphereMBS, SCNPyramidMBS, SCNConeMBS, SCNCylinderMBS, or SCNCapsuleMBS), SceneKit uses an idealized form of that geometry for the physics shape instead of using the geometry,Äôs vertex data to simulate collisions. For example, if you create a physics shape from an SCNSphere object, SceneKit simulates collisions for any object that passes within the sphere,Äôs radius.

Because the idealized forms of simple geometries are computationally much simpler than the vertex data needed for displaying them, using basic geometries for physics shapes (or compound shapes created from basic geometries with the shapeWithShapes method) often provides the best balance between simulation accuracy and performance.

To use the newly created physics shape, create a physics body with the the bodyWithType method, or assign the shape to the physicsShape property of an existing body.
See also:

- 13.39.4 Constructor(node as SCNNodeMBS, Options as Dictionary = nil) 1085

13.39.4 Constructor(node as SCNNodeMBS, Options as Dictionary = nil)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics shape from a node or hierarchy of nodes.
Notes: node: A node object. The node must contain an SCNGeometryMBS object in its geometry property or have one or more child (or descendant) nodes that contain geometry.
options: A dictionary of options affecting the level of detail of the physics shape, or nil to use default options. For applicable keys and their possible values, see Shape Creation Options Keys.

Returns a new physics shape object.

To use the newly created physics shape, create a physics body with the the bodyWithType method, or assign the shape to the physicsShape property of an existing body.

The node used to create the physics shape need not be the same as the node whose physics body you attach the shape to—or even be in the scene whose physics world you use the shape in. For example, you can create a physics body for a complex object by building a hierarchy of nodes containing simple geometries (using the SCNBoxMBS and SCNSphereMBS classes), and then creating a physics shape from those nodes. The resulting physics shape, a compound of bounding boxes or convex hulls, provides a rough approximation of the complex object without a high cost to simulation performance.
See also:

- 13.39.3 Constructor(geometry as SCNGeometryMBS, Options as Dictionary = nil) 1085

13.39.5 copy as SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the shape.

13.39.6 SCNPhysicsShapeKeepAsCompoundKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option for selecting whether to create a group of independent shapes or combine them into a single shape.
Notes: The value for this key is a boolean value. The default value is true, specifying that SceneKit convert separate geometries into separate shapes and join the resulting shapes. If false, SceneKit creates a single shape approximating the combined form of the geometries.

13.39.7 SCNPhysicsShapeOptionCollisionMargin as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Collision margin of the physics shape (as a number)

13.39.8 SCNPhysicsShapeScaleKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option for selecting the scale factor of the shape relative to the local coordinate space of the node containing it.
Notes: The value for this key is a SCNVector3MBS object , whose components describe the scale factor in

each of the x-, y- and z-axis directions. The default value is the vector { 1.0, 1.0, 1.0 } , specifying no change of scale.
SceneKit,Äôs physics simulation ignores the scale property of nodes containing physics bodies when simulating collisions. Instead, use this option to provide a scale factor when creating custom physics shapes. (If you create a physics body for a node without specifying a custom shape, SceneKit uses the node,Äôs scale property to infer this scale factor at creation time.)

13.39.9 SCNPhysicsShapeTypeBoundingBox as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics shape is the smallest box containing the geometry.
Notes: This option provides the lowest level of detail and the fastest simulation performance. Use it for generally box-shaped physics bodies or when constructing a compound physics shape.

13.39.10 SCNPhysicsShapeTypeConcavePolyhedron as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics shape is a concave polyhedron closely following the surface of the geometry.
Notes: This option provides the highest level of detail, at a high cost to simulation performance. Use it only for irregularly shaped bodies where precise collision behavior is crucial to your app,Äôs design.
This shape type may only be used for static physics bodies (that is, those whose type property is SCNPhysicsBodyTypeStatic).

13.39.11 SCNPhysicsShapeTypeConvexHull as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics shape is a convex polyhedron roughly enclosing the geometry.
Notes: This option provides a moderate level of detail and simulation performance. Use it for rounded or irregularly shaped physics bodies.

13.39.12 SCNPhysicsShapeTypeKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An option for selecting the level of detail at which to create shapes from geometry.
Notes: The value for this key is one of the constants listed in Shape Types. The default type is SCNPhysicsShapeTypeConvexHull.

13.39.13 shapeWithGeometry(geometry as SCNGeometryMBS, Options as Dictionary = nil) as SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics shape based on a geometry object.
Notes: geometry: A geometry object.
options: A dictionary of options affecting the level of detail of the physics shape, or nil to use default options. For applicable keys and their possible values, see Shape Creation Options Keys.

Returns a new physics shape object.

If you create a physics shape using one of the basic geometry classes (SCNBoxMBS, SCNSphereMBS, SCNPyramidMBS, SCNConeMBS, SCNCylinderMBS, or SCNCapsuleMBS), SceneKit uses an idealized form of that geometry for the physics shape instead of using the geometry,Äôs vertex data to simulate collisions. For example, if you create a physics shape from an SCNSphere object, SceneKit simulates collisions for any object that passes within the sphere,Äôs radius.

Because the idealized forms of simple geometries are computationally much simpler than the vertex data needed for displaying them, using basic geometries for physics shapes (or compound shapes created from basic geometries with the shapeWithShapes method) often provides the best balance between simulation accuracy and performance.

To use the newly created physics shape, create a physics body with the the bodyWithType method, or assign the shape to the physicsShape property of an existing body.

13.39.14 shapeWithNode(node as SCNNodeMBS, Options as Dictionary = nil) as SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a physics shape from a node or hierarchy of nodes.
Notes: node: A node object. The node must contain an SCNGeometryMBS object in its geometry property or have one or more child (or descendant) nodes that contain geometry.
options: A dictionary of options affecting the level of detail of the physics shape, or nil to use default options. For applicable keys and their possible values, see Shape Creation Options Keys.

Returns a new physics shape object.

To use the newly created physics shape, create a physics body with the the bodyWithType method, or assign the shape to the physicsShape property of an existing body.

The node used to create the physics shape need not be the same as the node whose physics body you attach the shape to—or even be in the scene whose physics world you use the shape in. For example, you can create a physics body for a complex object by building a hierarchy of nodes containing simple geometries (using the SCNBoxMBS and SCNSphereMBS classes), and then creating a physics shape from those nodes. The resulting physics shape, a compound of bounding boxes or convex hulls, provides a rough approximation of the complex object without a high cost to simulation performance.

13.39.15 shapeWithShapes(shapes() as SCNPhysicsShapeMBS, transforms() as SCNMatrix4MBS = nil) as SCNPhysicsShapeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new physics shape by combining others.
Notes: shapes: An array of SCNPhysicsShapeMBS objects.
transforms: An array SCNMatrix4MBS values, each of which is a transform for the physics shape at the corresponding index in the shapes parameter.

Returns a new physics shape object.

An individual physics shape is defined in its own local coordinate space. Therefore, to describe the positions and orientations of multiple shapes relative to one another, you must use coordinate transformations.

13.39.16 sourceObject as Variant

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The object that was used to create the shape.
Notes: This property, along with the transforms and options properties, provides the information that was used to create the shape. You can use this information, for example, to draw editing or debugging UI in your scene.

- If the shape was created with the shapeWithGeometry method, the source object is an SCNGeometryMBS object, and the options property contains the options affecting the shape,Äôs construction from that geometry.
- If the shape was created with the shapeWithNode method, the source object is an SCNNodeMBS object, and the options property contains the options affecting the shape,Äôs construction from that node.
- If the shape was created with the shapeWithShapes method, the source object is an array of SCNPhysicsShapeMBS objects and the transforms property describes how those shapes combine to form a compound shape.

13.39.17 transforms as SCNMatrix4MBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The array of transforms that was used to create a compound shape.
Notes: You provide this array of SCNMatrix4MBS values, in the shapeWithShapes method to create a compound shape. Use this array along with the sourceObject property to recover the information that was used to create the shape.

If the shape was created with the shapeWithGeometry or shapeWithNode method, this property,Äôs value is nil.

13.39.18 Properties

13.39.19 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.39.20 options as Dictionary

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The options dictionary that was used to create the shape.
Notes: You provide this dictionary in the shapeWithGeometry or shapeWithNode method. Use this dictionary along with the sourceObject property to recover the information that was used to create the shape. If the shape was created with the shapeWithShapes method, this property,Äôs value is nil.
(Read only property)

13.40 class SCNPhysicsSliderJointMBS

13.40.1 class SCNPhysicsSliderJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A physics behavior that connects two bodies and allows them to slide against each other and rotate around their connecting points.
Notes: A slider joint can have zero, one, or two degrees of freedom depending on whether you allow it to slide or rotate. You can also use a slider joint to pin a body so that it can move only by sliding a specific axis in the coordinate space of the node containing it. You can also use a slider joint as a motor, applying a force or torque to the bodies it connects.
Subclass of the SCNPhysicsBehaviorMBS class.

13.40.2 Methods

13.40.3 Constructor(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a slider joint that anchors a single physics body in space and allows it to slide along a specific axis.
Notes: body: The physics body to be controlled by the joint.
axis: The axis along which the first body can slide, relative to the node containing it.
anchor: The point at which the body is pinned, in the local coordinate system of the node containing it.

Returns a new slider joint behavior.

This method defines the location where the body is anchored in the coordinate system of the node containing it. To define its sliding or rotation motion relative to that point, use the properties listed in Limiting the Motion of a Slider Joint.
For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.40.4 Constructor(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) 1092

13.40.4 Constructor(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a slider joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.
axisA: The axis along which the first body can slide, relative to the node containing it.
anchorA: The point at which the joint connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.
axisB: The axis along which the second body can slide, relative to the node containing it.
anchorB: The point at which the joint connects, relative to the node containing the second body.

Returns a new slider joint behavior.

This method defines the location where the bodies are pinned together. To define their sliding or rotation motion relative to that point, use the properties listed in Limiting the Motion of a Slider Joint.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.40.3 Constructor(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) 1091

13.40.5 jointWithBody(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) as SCNPhysicsSliderJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a slider joint that anchors a single physics body in space and allows it to slide along a specific axis.
Notes: body: The physics body to be controlled by the joint.
axis: The axis along which the first body can slide, relative to the node containing it.
anchor: The point at which the body is pinned, in the local coordinate system of the node containing it.

Returns a new slider joint behavior.

This method defines the location where the body is anchored in the coordinate system of the node containing it. To define its sliding or rotation motion relative to that point, use the properties listed in Limiting the Motion of a Slider Joint.
For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your

scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.40.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) as SCNPhysicsSliderJointMBS 1093

13.40.6 jointWithBody(bodyA as SCNPhysicsBodyMBS, axisA as SCNVector3MBS, anchorA as SCNVector3MBS, bodyB as SCNPhysicsBodyMBS, axisB as SCNVector3MBS, anchorB as SCNVector3MBS) as SCNPhysicsSliderJointMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a slider joint connecting two physics bodies.
Notes: bodyA: The first physics body to be connected by the joint.
axisA: The axis along which the first body can slide, relative to the node containing it.
anchorA: The point at which the joint connects, relative to the node containing the first body.
bodyB: The second physics body to be connected by the joint.
axisB: The axis along which the second body can slide, relative to the node containing it.
anchorB: The point at which the joint connects, relative to the node containing the second body.

Returns a new slider joint behavior.

This method defines the location where the bodies are pinned together. To define their sliding or rotation motion relative to that point, use the properties listed in Limiting the Motion of a Slider Joint.

For a behavior to take effect, add it to the physics simulation by calling the addBehavior method on your scene,Äôs SCNPhysicsWorldMBS object. The physics bodies constrained by the joint must be attached to nodes in the scene.
See also:

- 13.40.5 jointWithBody(body as SCNPhysicsBodyMBS, axis as SCNVector3MBS, anchor as SCNVector3MBS) as SCNPhysicsSliderJointMBS 1092

13.40.7 Properties

13.40.8 anchorA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the joint connects, relative to the node containing the first body.
Notes: (Read and Write property)

13.40.9 anchorB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The point at which the joint connects, relative to the node containing the second body.
Notes: (Read and Write property)

13.40.10 axisA as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The axis along which the first body can slide, relative to the node containing it.
Notes: (Read and Write property)

13.40.11 axisB as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The axis along which the second body can slide, relative to the node containing it.
Notes: (Read and Write property)

13.40.12 bodyA as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The first physics body connected by the joint.
Notes: (Read only property)

13.40.13 bodyB as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The second physics body connected by the joint.
Notes: This property,Äôs value is nil if the joint was created using the jointWithBody method.
(Read only property)

13.40.14 maximumAngularLimit as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum rotation angle between the two bodies, measured in radians relative to their initial orientations.
Notes: The default (and maximum) value of this property is M_PI. With this value, the joint can spin counterclockwise (relative to the first body) with no limit.
Set both this property and the maximumAngularLimit property to the same value to prevent the bodies from rotating around their anchor points. (Set both properties to 0.0 to fix the bodies in their initial orientations.) Bodies whose orientation is fixed by a sliding joint may still slide, depending on the values of the minimumLinearLimit and maximumLinearLimit properties.
(Read and Write property)

13.40.15 maximumLinearLimit as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum distance between the anchor points of the two bodies, relative to their initial positions.
Notes: The default value of this property is INFINITY. With this value, the joint can slide forever in the direction of the slider axis.
Set both this property and the minimumLinearLimit property to the same value to pin the bodies together at their anchor points. (Set both properties to 0.0 to pin the bodies together at their initial positions.) Bodies pinned together by a sliding joint may still rotate, depending on the values of the minimumAngularLimit and maximumAngularLimit properties.
(Read and Write property)

13.40.16 minimumAngularLimit as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum rotation angle between the two bodies, measured in radians relative to their initial orientations.
Notes: The default (and minimum) value of this property is -M_PI. With this value, the joint can spin clockwise (relative to the first body) with no limit.
Set both this property and the maximumAngularLimit property to the same value to prevent the bodies from rotating around their anchor points. (Set both properties to 0.0 to fix the bodies in their initial orientations.) Bodies whose orientation is fixed by a sliding joint may still slide, depending on the values of the minimumLinearLimit and maximumLinearLimit properties.
(Read and Write property)

13.40.17 minimumLinearLimit as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum distance between the anchor points of the two bodies, relative to their initial positions.
Notes: The default value of this property is -INFINITY. With this value, the joint can slide forever in the direction opposite the slider axis.
Set both this property and the maximumLinearLimit property to the same value to pin the bodies together at their anchor points. (Set both properties to 0.0 to pin the bodies together at their initial positions.) Bodies pinned together by a sliding joint may still rotate, depending on the values of the minimumAngularLimit and maximumAngularLimit properties.
(Read and Write property)

13.40.18 motorMaximumForce as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum linear force that the joint can apply to its connected bodies, in newtons.
Notes: When you change the value of the motorTargetLinearVelocity property, the joint continuously applies a force of no greater than this magnitude until the bodies are moving at the target velocity. The default value is 1.0.
(Read and Write property)

13.40.19 motorMaximumTorque as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum torque that the joint can apply to its connected bodies, in newton-meters.
Notes: When you change the value of the motorTargetAngularVelocity property, the joint continuously applies a force of no greater than this magnitude until the bodies are rotating around the joint at the target angular velocity. The default value is 1.0.
(Read and Write property)

13.40.20 motorTargetAngularVelocity as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The angular velocity at which the joint,Äôs connected bodies should rotate around it.
Notes: At the default value of 0.0, the joint moves only when an external force acts on one of its connected bodies. Changing this value causes the joint to act as a rotary motor, continuously applying a torque (specified by the motorMaximumTorque property) until its connected bodies are rotating around the joint at the new angular velocity.

(Read and Write property)

13.40.21 motorTargetLinearVelocity as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The velocity at which the joint,Äôs connected bodies should slide.
Notes: At the default value of 0.0, the joint moves only when an external force acts on one of its connected bodies. Changing this value causes the joint to act as a linear motor, continuously applying a force (specified by the motorMaximumForce property) until its connected bodies are moving along the sliding axis of the joint at the new velocity.
(Read and Write property)

13.41 class SCNPhysicsVehicleMBS

13.41.1 class SCNPhysicsVehicleMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A physics behavior that modifies a physics body to behave like a car, motorcycle, or other wheeled vehicle.
Notes: To build a vehicle, designate an SCNPhysicsBodyMBS object as its chassis and an array of SCNPhysicsVehicleWheelMBS objects as its wheels. For each wheel, you define physical characteristics such as suspension and traction, and associate a node in your scene to provide the wheel,Äôs size and visual representation. After you construct a vehicle, you can control it in terms of acceleration, braking, and steering.

Although it,Äôs also possible to use a set of physics bodies and joints to collectively simulate a wheeled vehicle, the SCNPhysicsVehicleMBS class implements a higher-level simulation that provides realistic vehicle behavior with more efficient simulation performance.
Subclass of the SCNPhysicsBehaviorMBS class.

13.41.2 Methods

13.41.3 applyBrakingForce(value as double, index as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Applies a force between the specified wheel and the ground under the vehicle.
Notes: value: The magnitude of the torque, in newton-meters.
index: The index of the wheel applying the force.

Applying a braking force causes the wheel to slow down regardless of the direction it,Äôs currently spinning in.

As with all physical quantities in SceneKit, you need not use realistic force measurements in your app—the effects of the physics simulation depend on the relative differences between forces, not on their absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.

Calling this method applies a braking force for one step (or frame) of the physics simulation. To continuously decelerate a vehicle, call this method again on subequent simulation steps (for example, from your scene renderer delegate,Äôs renderer method) until the vehicle stops or reaches your desired speed.

13.41.4 applyEngineForce(value as double, index as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Applies a force between the specified wheel and the ground under the vehicle.
Notes: value: The magnitude of the force, in newtons.
index: The index of the wheel applying the force.

Applying a positive force turns the wheel in a direction that would move the vehicle forward; applying a negative force moves the vehicle in reverse.

As with all physical quantities in SceneKit, you need not use realistic force measurements in your app—the effects of the physics simulation depend on the relative differences between forces, not on their absolute values. You may use whatever values produce the behavior or gameplay you,Äôre looking for as long as you use them consistently.

Calling this method applies a force for one step (or frame) of the physics simulation. To continuously accelerate a vehicle, call this method again on subequent simulation steps (for example, from your scene renderer delegate,Äôs renderer method) until the vehicle reaches your desired speed.

13.41.5 Constructor(chassisBody as SCNPhysicsBodyMBS, wheels() as SCNPhysicsVehicleWheelMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a vehicle behavior.
Notes: chassisBody: A physics body to serve as the vehicle,Äôs chassis.
wheels: An array of SCNPhysicsVehicleWheel objects representing the vehicle,Äôs wheels. A vehicle must have at least one wheel.

Returns a new vehicle behavior.

Each object in the wheels array associates a node with the wheel to serve as its visual representation and defines properties for the wheel,Äôs physical characteristics. Each wheel object must reference a unique node, which should be a child of the node containing the physics body used for the vehicle,Äôs chassis. Typically, you load a node hierarchy representing the vehicle and all of its wheels from a scene file and then designate which nodes serve as the body and wheels.

For a behavior to take effect, you must add it to the physics simulation by calling the addBehavior: method on your scene,Äôs SCNPhysicsWorldMBS object.

13.41.6 setSteeringAngle(value as double, index as Integer)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Pivots the specified wheel around its steering axis.
Notes: value: The angle to set the wheel at relative to its steering axis, in radians.
index: The index, in the vehicle,Äôs wheels array, of the wheel to be pivoted.

Steering angles are relative to the wheel,Äôs steeringAxis vector. With the default steering axis of { 0.0, -1.0, 0.0 } , a steering angle of 0.0 represents neutral steering, positive values steer the vehicle to the right, and negative values steer to the left.

13.41.7 vehicleWithChassisBody(chassisBody as SCNPhysicsBodyMBS, wheels() as SCNPhysicsVehicleWheelMBS) as SCNPhysicsVehicleMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a vehicle behavior.
Notes: chassisBody: A physics body to serve as the vehicle,Äôs chassis.
wheels: An array of SCNPhysicsVehicleWheel objects representing the vehicle,Äôs wheels. A vehicle must have at least one wheel.

Returns a new vehicle behavior.

Each object in the wheels array associates a node with the wheel to serve as its visual representation and defines properties for the wheel,Äôs physical characteristics. Each wheel object must reference a unique node, which should be a child of the node containing the physics body used for the vehicle,Äôs chassis. Typically, you load a node hierarchy representing the vehicle and all of its wheels from a scene file and then designate which nodes serve as the body and wheels.

For a behavior to take effect, you must add it to the physics simulation by calling the addBehavior: method on your scene,Äôs SCNPhysicsWorldMBS object.

13.41.8 wheels as SCNPhysicsVehicleWheelMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: An array of SCNPhysicsVehicleWheelMBS objects representing the vehicle,Äôs wheels.
Notes: You can dynamically change the suspension and traction properties of a wheel connected to the vehicle by using the corresponding SCNPhysicsVehicleWheelMBS object.

13.41.9 Properties

13.41.10 chassisBody as SCNPhysicsBodyMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics body representing the vehicle,Äôs chassis.
Notes: The vehicle,Äôs chassis must be a dynamic body.
(Read only property)

13.41.11 speedInKilometersPerHour as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The vehicle,Äôs ground speed, in kilometers per hour.
Notes: (Read only property)

13.42 class SCNPhysicsVehicleWheelMBS

13.42.1 class SCNPhysicsVehicleWheelMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The appearance and physical characteristics of an individual wheel associated with an physics vehicle behavior.
Notes: To use wheels in a vehicle simulation, include them when creating an SCNPhysicsVehicle object with the vehicleWithChassisBody / Constructro initializer, then add the vehicle object to your scene,Äôs physics world using the physics world,Äôs addBehavior method.

Creating a Wheel
You create a wheel with an SCNNodeMBS object whose contents provide the wheel,Äôs visual representation—a geometry that rotates when the simulated vehicle rolls along a surface. The node representing a wheel must be a child of the node containing the physics body that serves as the vehicle,Äôs chassis, and each wheel in a vehicle must reference a unique node. Typically, you load a scene file that contains a node hierarchy representing the vehicle and all of its wheels. Next, you designate which nodes serve as the body and wheels.
Because the SCNPhysicsVehicleMBS behavior that a wheel is attached to manages its participation in the physics simulation, you don,Äôt need to attach a physics body to the SCNNodeMBS object representing a wheel.

Changing a Wheel,Äôs Physical Properties
The properties of a wheel define the geometry of its connection to the vehicle and simulate its size, traction, and suspension. You can change these properties after the wheel and the vehicle containing it have been added to the physics world. In this way, you can simulate effects such as variable suspension and flat tires.

Note
Vehicles and their wheels have several properties measured in real-world units (meters, centimeters, and newtons) with default values that produce realistic behavior for vehicles of size similar to an average automobile. If you design your scene on a different scale, proportionally change the values of these properties to fit the desired behavior of your app or game.

13.42.2 Methods

13.42.3 Constructor(node as SCNNodeMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a wheel object.
Notes: node: The node whose contents provide the wheel,Äôs visual representation.

Returns a new wheel object.

The node representing a wheel must be a child of the node whose physics body serves as the chassis of the SCNPhysicsVehicleMBS behavior the wheel is attached to. Each wheel object must reference a unique node. To use the wheel, add it to the vehicle behavior using the addWheel method.
SceneKit uses the node,Äôs bounding box to determine the wheel,Äôs initial size, and it uses the node,Äôs position to determine the where the wheel connects to the vehicle,Äôs chassis. You can change attributes using the radius and connectionPosition properties.

13.42.4 copy as SCNPhysicsVehicleWheelMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the wheel.

13.42.5 wheelWithNode(node as SCNNodeMBS) as SCNPhysicsVehicleWheelMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a wheel object.
Notes: node: The node whose contents provide the wheel,Äôs visual representation.

Returns a new wheel object.

The node representing a wheel must be a child of the node whose physics body serves as the chassis of the SCNPhysicsVehicleMBS behavior the wheel is attached to. Each wheel object must reference a unique node. To use the wheel, add it to the vehicle behavior using the addWheel method.
SceneKit uses the node,Äôs bounding box to determine the wheel,Äôs initial size, and it uses the node,Äôs position to determine the where the wheel connects to the vehicle,Äôs chassis. You can change attributes using the radius and connectionPosition properties.

13.42.6 Properties

13.42.7 Axle as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The direction of the axis that the wheel spins around to move the vehicle.
Notes: This vector is expressed in the coordinate space of the node containing the vehicle,Äôs chassis. The default axle direction is { -1.0, 0.0, 0.0 } .

(Read and Write property)

13.42.8 ConnectionPosition as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The position of the wheel,Äôs connection to the vehicle,Äôs chassis.
Notes: This vector is expressed in the coordinate space of the node containing the vehicle,Äôs chassis. When you create a wheel from a node, SceneKit uses the node,Äôs position property as the wheel,Äôs connection point.
(Read and Write property)

13.42.9 FrictionSlip as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The traction between the wheel and any surface in contact with it.
Notes: The default value of this property is 1.0. Lower values result in better traction, and higher values make the wheel more likely to slip (causing it to spin freely instead of moving the vehicle).
(Read and Write property)

13.42.10 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.42.11 MaximumSuspensionForce as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum force of the suspension between the vehicle and the wheel, in newtons.
Notes: The physics simulation applies a force of no greater than this magnitude when contact with the ground causes the wheel to move relative to the vehicle. The default maximum suspension force is 6000.0.
(Read and Write property)

13.42.12 MaximumSuspensionTravel as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum distance that the wheel is allowed to move up or down relative to its connection point, in centimeters.
Notes: Travel is the total distance a wheel is allowed to move (in both directions), in the coordinate system of the node containing the vehicle,Äôs chassis. The default suspension travel is 500.0.
(Read and Write property)

13.42.13 Node as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The node providing the wheel,Äôs visual representation.
Notes: SceneKit automatically rotates and repositions this node in response to the physics simulation.
(Read only property)

13.42.14 Radius as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The radius of the wheel.
Notes: When you create a wheel from a node, its default radius is half of the largest dimension of the node,Äôs bounding box. (A wheel is always circular, even if the content of the node representing it is not.)
(Read and Write property)

13.42.15 SteeringAxis as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The direction of the axis that the wheel pivots around to steer the vehicle.
Notes: This vector is expressed in the coordinate space of the node containing the vehicle,Äôs chassis. The default steering axis is { 0.0, -1.0, 0.0 } .
When you steer a wheel using the vehicle,Äôs setSteeringAngle() method, the wheel pivots relative to this axis. For example, you can implement a vehicle whose rear wheels steer opposite its front wheels by reversing this vector,Äôs direction for the rear wheels and then applying the same steering angle to all wheels.
(Read and Write property)

13.42.16 SuspensionCompression as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The coefficient that limits the speed of the suspension returning to its rest length when compressed.
Notes: The default suspension coefficient is 4.4. Lower values cause the wheel to return to its natural position more quickly.
(Read and Write property)

13.42.17 SuspensionDamping as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The damping ratio that limits oscillation in the vehicle,Äôs suspension.
Notes: Damping ratio measures the tendency of the suspension to oscillate after a shock—in other words, for the vehicle to bounce up and down after running over a bump. The default damping ratio of 2.3 causes the wheel to return to its neutral position quickly after a shock. Values lower than 1.0 result in more oscillation.
(Read and Write property)

13.42.18 SuspensionRestLength as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The resting length of the suspension, in meters.
Notes: This property measures the length of the simulated spring between the vehicle and its wheel when the spring is not stressed by the weight of either body. When the wheel receives a shock (for example, when the vehicle runs over a bump), SceneKit adds the difference between the wheel,Äôs current position and its connection position to this rest length and then applies a force between the wheel and vehicle proportional to the total.
(Read and Write property)

13.42.19 SuspensionStiffness as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The spring coefficient of the suspension between the vehicle and the wheel.
Notes: The spring coefficient determines both how quickly the wheel returns to its natural position after a shock (for example, when the vehicle runs over a bump) and how much force from the shock it transmits to the vehicle. The default spring coefficient is 2.0.
(Read and Write property)

13.43 class SCNPhysicsWorldMBS

13.43.1 class SCNPhysicsWorldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The global simulation of collisions, gravity, joints, and other physics effects in a scene.
Notes: You do not create SCNPhysicsWorldMBS objects directly; instead, read the physicsWorld property of an SCNSceneMBS object. Use physics world object to perform the following tasks:

- Manage global properties of the simulation, such as its speed and constant gravity. (For more precise control of gravity and similar effects, see the SCNPhysicsFieldMBS class.)
- Register behaviors that modify interactions between the scene,Äôs physics bodies, such as joints and vehicles. For more details, see SCNPhysicsBehaviorMBS.
- Specify a delegate object to receive messages when two physics bodies contact each other
- Perform specific contact tests, and search for physics bodies in the scene using ray and sweep tests.

Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.2
- MBS Xojo Plugins, version 20.2pr2

Videos

- XDC 2020 MBS Plugins Presentation

13.43.2 Methods

13.43.3 addBehavior(behavior as SCNPhysicsBehaviorMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a behavior to the physics world.
Notes: behavior: The behavior to be added.

Physics behaviors constrain or modify the effects of the physics simulation on sets of physics bodies. For example, the SCNPhysicsHingeJointMBS behavior causes two bodies to move as if connected by a hinge that pivots around a specific axis, and the SCNPhysicsVehicleMBS behavior causes a body to roll like a car or other wheeled vehicle.

To use a behavior in your scene, follow these steps:
1. Create SCNPhysicsBody objects and attach them to each node that participates in the behavior.

2. Create and configure a behavior object joining the physics bodies. See SCNPhysicsBehaviorMBS for a list of behavior classes.
3. Call addBehavior on your scene,Äôs physics world object to add the behavior to the physics simulation.

13.43.4 allBehaviors as SCNPhysicsBehaviorMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The list of behaviors affecting bodies in the physics world.

13.43.5 Constructor(fireContactEvents as Boolean = false)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.
Notes:

If fireContactEvents is true or you call EnableContactsEvents, we enable the contact events.

A contact is created when two physics bodies overlap and one of the physics bodies has a collisionBitMask property that overlaps with the other body,Äôs categoryBitMask property.

13.43.6 contactTest(body as SCNPhysicsBodyMBS, options as Dictionary = nil) as SCNPhysicsContactMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks for contacts between one physics body and any other bodies in the physics world.
Notes: body: The body to test for contact.
options: A dictionary of options affecting the test, or nil to use default options. For applicable keys and the possible values, see Physics Test Options Keys.

Returns an array of SCNPhysicsContactMBS objects describing contacts between the specified body and any others, or nil if the body is not in contact with any other bodies.

SceneKit sends messages to the physics world,Äôs contactdelegate object only when collisions occur between bodies whose collisionBitMask and categoryBitMask properties overlap, and only for collisions between certain types of bodies. (For details, see SCNPhysicsBodyType.) Use this method to directly test for all contacts between one body and any other bodies at a time of your choosing. For example, to implement a game with a ,Äúwall jump,Äù effect, you could call this method when the player presses the jump button to see if the player character is in contact with any walls.

See also:

- 13.43.7 contactTest(bodyA as SCNPhysicsBodyMBS, bodyB as SCNPhysicsBodyMBS, options as Dictionary = nil) as SCNPhysicsContactMBS() 1109

13.43.7 contactTest(bodyA as SCNPhysicsBodyMBS, bodyB as SCNPhysicsBodyMBS, options as Dictionary = nil) as SCNPhysicsContactMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Checks for contacts between two physics bodies.
Notes: bodyA: The first body (to test for contact with the second).
bodyB: The second body (to test for contact with the first).
options: A dictionary of options affecting the test, or nil to use default options. For applicable keys and the possible values, see Physics Test Options Keys.

Returns an array of SCNPhysicsContactMBS objects describing contacts between the two bodies, or nil if the bodies are not in contact.

SceneKit sends messages to the physics world,Äôs contactDelegate object only when collisions occur between bodies whose collisionBitMask and categoryBitMask properties overlap, and only for collisions between certain types of bodies. (For details, see SCNPhysicsBodyType.) Use this method to directly test for contacts between any two bodies at a time of your choosing. For example, to implement a game where the player character can pick up an item, you might call this method when the player presses the ,Äúpick up,Äù button to see if the player character is in contact with the item to be picked up.
See also:

- 13.43.6 contactTest(body as SCNPhysicsBodyMBS, options as Dictionary = nil) as SCNPhysicsContactMBS() 1108

13.43.8 convexSweepTest(shape as SCNPhysicsShapeMBS, fromTransform as SCNMatrix4MBS, toTransform as SCNMatrix4MBS, options as Dictionary = nil) as SCNPhysicsContactMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Searches for physics bodies in the space formed by moving a convex shape through the physics world.
Notes: shape: A physics shape. This shape must enclose a convex volume. For details on creating shapes that satisfy this requirement, see SCNPhysicsShapeMBS.
from: A transform matrix representing the initial position and orientation of the shape.
to: A transform matrix representing the final position and orientation of the shape.
options: A dictionary of options affecting the test, or nil to use default options. For applicable keys and the possible values, see Physics Test Options Keys.

Returns an array of SCNPhysicsContactMBS objects describing any contacts that would occur when moving the physics shape through the physics world.

Use this method when it,Äôs important to plan for (or avoid) collisions ahead of the physics simulation. For example, in a game you might plan maneuvers for a flying character to fit through the gaps between static bodies in the physics world

13.43.9 EnableContactsEvents

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Enables contact events.
Notes: If you connect the contact events via AddHandler, please call this method once to connect the events internally.

13.43.10 rayTestWithSegment(origin as SCNVector3MBS, dest as SCNVector3MBS, options as Dictionary = nil) as SCNPhysicsBehaviorMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Searches for physics bodies along a line segment between two points in the physics world.
Notes: origin: An endpoint of the line segment to search, specified in the scene,Äôs world coordinate system.
dest: The other endpoint of the line segment to search, specified in the scene,Äôs world coordinate system.
options: A dictionary of options affecting the test, or nil to use default options. For applicable keys and the possible values, see Physics Test Options Keys.

Returns an array of SCNHitTestResultMBS objects describing search results.

Use this method to implement concepts such as line of sight in your app. For example, in a game you might implement behavior for an enemy character by searching for physics bodies along a line between the enemy character,Äôs position and the player character,Äôs position

13.43.11 removeAllBehaviors

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes all behaviors affecting bodies in the physics world.

13.43.12 removeBehavior(behavior as SCNPhysicsBehaviorMBS)

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes a behavior from the physics world.

13.43.13 SCNPhysicsTestBackfaceCullingKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The key for choosing whether to ignore back-facing polygons in physics shapes when searching for contacts.
Notes: The value for this key is a Boolean value. The default value is true, specifying that the search should only return contacts with the exterior surfaces of any physics shapes. Change the value to false to consider contacts with both interior and exterior surfaces.

This key applies only to ray and convex sweep tests, and only to physics shapes created using the SCNPhysicsShapeTypeConcavePolyhedron option.

13.43.14 SCNPhysicsTestCollisionBitMaskKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The key for selecting which categories of physics bodies that SceneKit should test for contacts.
Notes: The value for this key is an unsigned Integer value. SceneKit tests for contacts only with physics bodies whose categoryBitMask property overlaps with this bit mask. The default value is SCNPhysicsCollisionCategoryAll, specifying that searches should test all physics bodies regardless of their category.

13.43.15 SCNPhysicsTestSearchModeAll as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the option values for SCNPhysicsTestSearchModeKey key.
Notes: Searches should return all contacts matching the search parameters.

13.43.16 SCNPhysicsTestSearchModeAny as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the option values for SCNPhysicsTestSearchModeKey key.
Notes: Searches should return only the first contact found regardless of its position relative to the search

parameters.

13.43.17 SCNPhysicsTestSearchModeClosest as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the option values for SCNPhysicsTestSearchModeKey key.
Notes: Searches should return only the closest contact to the beginning of the search.

13.43.18 SCNPhysicsTestSearchModeKey as String

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The key for selecting the number and order of contacts to be tested.
Notes: See Physics Test Search Modes for possible values. The default value is SCNPhysicsTestSearchModeAny.
This key applies only to ray and convex sweep tests.

13.43.19 updateCollisionPairs

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Forces the physics engine to reevaluate possible collisions between physics bodies.
Notes: By default, SceneKit checks for collisions between physics bodies only once per simulation step. If you directly change the positions of any physics bodies outside of a event, call the updateCollisionPairs method before using any of the methods listed in Searching for Physics Bodies Detecting Contacts Between Physics Bodies.

13.43.20 Properties

13.43.21 gravity as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A vector that specifies the gravitational acceleration applied to physics bodies in the physics world.
Notes: The components of this vector are measured in meters per second per second. The default value is (0.0,-9.8,0.0).
This property applies a constant acceleration to all physics bodies in the world, simulating the effect of gravity near the surface of the Earth. For more sophisticated gravity effects, including limited areas of effect and strength proportional to distance, use the SCNPhysicsFieldMBS class. When using fields, you may want

to set this property to the zero vector so that fields provide all gravity effects in the physics world.
(Read and Write property)

13.43.22 Handle as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Internal object reference.
Notes: (Read and Write property)

13.43.23 speed as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The rate at which the simulation executes.
Notes: The default value is 1.0, which means that the simulation runs at normal speed. A value other than the default changes the rate at which time passes in the physics simulation. For example, a speed value of 2.0 indicates that time in the physics simulation passes twice as fast as the scene,Äôs simulation time. A value of 0.0 pauses the physics simulation.

Note:
Increasing the speed of the physics simulation reduces its accuracy.
(Read and Write property)

13.43.24 timeStep as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The time interval between updates to the physics simulation.
Notes: SceneKit processes the physics simulation and updates the state of all physics bodies once per the time interval specified by this property. The default value is 1/60 second (a rate of 60 Hz).
A faster simulation rate provides more accuracy in simulation results—such as collisions between fast-moving objects—but at a higher cost in CPU time (which may in turn slow down your app,Äôs rendering frame rate). Typically, you should set this property to match your target rendering frame rate (as defined by the preferredFramesPerSecond property of the SCNViewMBS object rendering your scene).
(Read and Write property)

13.43.25 Events

13.43.26 didBeginContact(contact as SCNPhysicsContactMBS)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the class that two bodies have come into contact.
Notes: contact: An object that describes the contact.

13.43.27 didEndContact(contact as SCNPhysicsContactMBS)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the class that a contact has ended.
Notes: contact: An object that describes the contact.

13.43.28 didUpdateContact(contact as SCNPhysicsContactMBS)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Tells the class that new information is available about an ongoing contact.
Notes: contact: An object that describes the contact.

SceneKit calls this method on each step of the physics simulation (see the timeStep property) if information about the contact changes—for example, if two bodies are sliding against one another.

13.44 class SCNPlaneMBS

13.44.1 class SCNPlaneMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNPlaneMBS represents a rectangle with controllable width and height. The plane has one visible side.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 28: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.44.2 Methods

13.44.3 Constructor(Width as Double, Height as Double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a plane with given width and height.

13.44.4 plane(Width as Double, Height as Double) as SCNPlaneMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a plane with given width and height.

13.44.5 Properties

13.44.6 cornerRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The corner radius. Animatable.
Notes: If the value is strictly less than 0, the geometry is empty. The default value is 0.
(Read and Write property)

13.44.7 cornerSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions for the rounded corners. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 10.
(Read and Write property)

13.44.8 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The plane extent along the Y axis. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.44.9 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. The default value is 1. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.44.10 width as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The plane extent along the X axis. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.44.11 widthSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the X axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.45 class SCNPyramidMBS

13.45.1 class SCNPyramidMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNPyramid represents a right pyramid with a rectangular base.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- Physics example for SceneKit in Xojo
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 29: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.45.2 Methods

13.45.3 Constructor(width as double, height as double, length as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a pyramid with given width, height, and length.
Notes: width: The width of the pyramid.
height: The height of the pyramid.
length: The length of the pyramid.

13.45.4 pyramid(width as double, height as double, length as double) as SCNPyramidMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a pyramid with given width, height, and length.
Notes: width: The width of the pyramid.
height: The height of the pyramid.
length: The length of the pyramid.

13.45.5 Properties

13.45.6 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the pyramid. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.45.7 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.45.8 length as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The length of the pyramid base. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.45.9 lengthSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Z axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.45.10 width as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The width of the pyramid base. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.

(Read and Write property)

13.45.11 widthSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the X axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.46 class SCNReplicatorConstraintMBS

13.46.1 class SCNReplicatorConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNReplicatorConstraint replicates the position/orientation/scale of a target node
Notes: Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.46.2 Methods

13.46.3 Constructor(target as SCNNodeMBS)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNReplicatorConstraint constraint.

13.46.4 replicatorConstraintWithTarget(target as SCNNodeMBS) as SCNReplicatorConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNReplicatorConstraint constraint.

13.46.5 Properties

13.46.6 orientationOffset as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines an addition orientation offset.
Notes: Defaults to no offset. Animatable.
(Read and Write property)

13.46.7 positionOffset as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines an addition orientation offset.
Notes: Defaults to no offset. Animatable.
(Read and Write property)

13.46.8 replicatesOrientation as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines whether or not the constraint should replicate the target orientation.
Notes: Defaults to true.
(Read and Write property)

13.46.9 replicatesPosition as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines whether or not the constraint should replicate the target position.
Notes: Defaults to true.
(Read and Write property)

13.46.10 replicatesScale as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines whether or not the constraint should replicate the target scale.
Notes: Defaults to true.
(Read and Write property)

13.46.11 scaleOffset as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines an addition scale offset.
Notes: Defaults to no offset. Animatable.
(Read and Write property)

13.46.12 target as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the target node to replicate.
Notes: (Read and Write property)

13.47 class SCNSceneMBS

13.47.1 class SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A container for the node hierarchy and global properties that together form a displayable 3D scene.
Notes: To display 3D content with SceneKit, you create a scene containing a hierarchy of the nodes and attributes that together represent your visual elements. Typically, you build your assets in a 3D visual editor, then assemble them into a scene using Xcode,Äôs SceneKit Scene Editor, ready for SceneKit to render.

The simplest way to create a scene is through Xcode,Äôs SceneKit Scene Editor. Start by importing one or more assets from a 3D editor, such as Blender. Then you adjust the positions and attributes of the assets, and set global scene properties, such as lighting environment, to compose your scene. The scene editor creates a .scn file, which you save to a .scnassets folder in the app bundle. When you build your project, Xcode optimizes the scene file for your target platform.
Blog Entries

- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 19.6, pages 45 to 46: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, page 53: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.4, page 39: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, pages 26 to 27: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.47.2 Methods

13.47.3 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an empty scene.

13.47.4 scene as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns an empty scene.
Notes: An empty scene contains only a root SCNNode object with no contents. To populate the scene, add children to the root node.

13.47.5 sceneNamed(name as string) as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Loads a scene from a file with the specified name in the app,Äôs main bundle.
Notes: name: The name of a scene file in the app bundle,Äôs resources directory.

Returns a new scene object, or nil if no scene could be loaded.

This method provides a convenient way to load a complete scene from a file in the app,Äôs main bundle. Calling this method is equivalent to using the NSBundle class to locate the scene file and passing the resulting URL to the sceneWithURL method, specifying no options and no error handling.

For more detailed options or to load only part of a file,Äôs scene graph, use the SCNSceneSource class.

When creating a scene using Xcode's Scene Editor or an external tool, you should copy your scene file into a directory with the .scnassets extension inside your app bundle. You should also place any image files referenced as textures from that scene in an Asset Catalog. Xcode will optimize the scene and texture resources for best performance on each target device, and prepare your texture resources for delivery features such as App Thinning and On-Demand Resources.
See also:

- 13.47.6 sceneNamed(name as string, folder as folderItem, options as Dictionary = nil) as SCNSceneMBS 1125

13.47.6 sceneNamed(name as string, folder as folderItem, options as Dictionary = nil) as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Loads a scene from a file with the specified name in a specific subdirectory of the app,Äôs main bundle.
Notes: name: The name of a scene file in the app bundle.
directory: The path to the subdirectory of the bundle,Äôs resources directory containing the scene file.
options: A dictionary of options affecting scene loading, or nil for default options. For available keys, see Scene Loading Options.

Returns a new scene object, or nil if no scene could be loaded.

This method provides a convenient way to load a complete scene from a file in the app,Äôs main bundle. Calling this method is equivalent to using the NSBundle class to locate the scene file and passing the resulting URL to the sceneWithURL method.

For more detailed options or to load only part of a file,Äôs scene graph, use the SCNSceneSource class.

When creating a scene using Xcode's Scene Editor or an external tool, you should copy your scene file into a directory with the .scnassets extension inside your app bundle. You should also place any image files referenced as textures from that scene in an Asset Catalog. Xcode will optimize the scene and texture resources for best performance on each target device, and prepare your texture resources for delivery features such as App Thinning and On-Demand Resources.
See also:

- 13.47.5 sceneNamed(name as string) as SCNSceneMBS 1125

13.47.7 sceneWithFile(file as folderItem, options as Dictionary = nil, byref Error as NSErrorMBS) as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Loads a scene from the specified file.
Notes: file: The scene file to load.
options: A dictionary of options affecting scene loading, or nil for default options. For available keys, see Scene Loading Options.
error: If an error occurs, this pointer is set to an NSError object describing the error. If you do not want error information, pass in nil.

Returns a new scene object, or nil if no scene could be loaded.

This method provides a convenient way to load a complete scene from a file at an arbitrary URL. For more detailed options or to load only part of a file,Äôs scene graph, use the SCNSceneSource class.

When creating a scene using Xcode's Scene Editor or an external tool, you should copy your scene file into a directory with the .scnassets extension inside your app bundle. You should also place any image files referenced as textures from that scene in an Asset Catalog. Xcode will optimize the scene and texture resources for best performance on each target device, and prepare your texture resources for delivery features such as App Thinning and On-Demand Resources.

13.47.8 sceneWithURL(FileURL as string, options as Dictionary = nil, byref Error as NSErrorMBS) as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Loads a scene from the specified URL.
Notes: url: The URL to the scene file to load.
options: A dictionary of options affecting scene loading, or nil for default options. For available keys, see Scene Loading Options.
error: If an error occurs, this pointer is set to an NSError object describing the error. If you do not want error information, pass in nil.

Returns a new scene object, or nil if no scene could be loaded.

This method provides a convenient way to load a complete scene from a file at an arbitrary URL. For more detailed options or to load only part of a file,Äôs scene graph, use the SCNSceneSource class.

When creating a scene using Xcode's Scene Editor or an external tool, you should copy your scene file into a directory with the .scnassets extension inside your app bundle. You should also place any image files referenced as textures from that scene in an Asset Catalog. Xcode will optimize the scene and texture resources for best performance on each target device, and prepare your texture resources for delivery features such as App Thinning and On-Demand Resources.

13.47.9 SCNSceneEndTimeAttributeKey as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the constant name for the attributes.
Notes: A floating-point value for the end time of the scene.

13.47.10 SCNSceneExportDestinationURL as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The final destination URL (an NSURLMBS object) for the exported scene file.
Notes: Use this option if you export a scene to a temporary directory and then move it to a final location. You must specify a final destination URL if your scene references external resources, such as image files for textures. SceneKit uses this URL to construct appropriate paths for external resources when writing the scene file.

13.47.11 SCNSceneFrameRateAttributeKey as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the constant name for the attributes.
Notes: A floating-point value (in an NSNumber object) for the frame rate of the scene.

This value may be present in scenes loaded from scene files produced using external tools, but has no effect on SceneKit,Äôs rendering of the scene.

13.47.12 SCNSceneStartTimeAttributeKey as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the constant name for the attributes.
Notes: A floating-point value for the start time of the scene.

13.47.13 SCNSceneUpAxisAttributeKey as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: One of the constant name for the attributes.
Notes: An SCNVector3MBS object (in an NSValueMBS object) specifying the orientation of the scene.

This value may be present in scenes loaded from scene files produced using external tools, but has no effect on SceneKit,Äôs processing of the scene. Use this vector when combining elements from different scenes so that they appear in their expected orientation.

13.47.14 Properties

13.47.15 background as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A background to be rendered before the rest of the scene.
Notes: If the material property,Äôs contents object is nil, SceneKit does not draw any background before drawing the rest of the scene. (If the scene is presented in an SCNView instance, the view,Äôs background color is visible behind the contents of the scene.)
If you specify a cube map texture for the material property (see the discussion of the contents property), SceneKit renders the background as a skybox.
(Read only property)

13.47.16 fogColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The color of the fog effect to be rendered with the scene. Animatable.
Notes: The default fog color is white.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.47.17 fogDensityExponent as Double

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The transition curve for the fog,Äôs intensity between its start and end distances. Animatable.
Notes: A fog effect fades out the contents of the scene with increasing distance from the pointOfView location, replacing them with increasing intensities of the fogColor color. The fogDensityExponent property determines the smoothness or abruptness of this transition.

A value of 0.0 (the default) specifies no attenuation—the fog,Äôs intensity is the same at all distances . A value of 1.0 specifies a linear transition, and a value of 2.0 specifies a quadratic transition curve. Higher values have little visible effect.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.47.18 fogEndDistance as Double

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The distance from a point of view at which the scene,Äôs contents are completely obscured by fog. Animatable.
Notes: A fog effect causes scene contents to become less visible the farther they are from the pointOfView node currently used for rendering. At distances less than the value of the fogStartDistance property, scene contents are fully visible. At greater distances, SceneKit blends the rendered scene contents with a constant color (specified by the fogColor property). At distances greater than the fogEndDistance property, the scene contents fade away completely and only the fog color is visible. Use fog to add atmospheric effects to your app or game, or to improve rendering performance by hiding parts of the scene that are far away from the current point of view.

The default end distance of 0.0 disables the fog effect. Change this property,Äôs value to enable fog.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.47.19 fogStartDistance as Double

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The distance from a point of view at which the scene,Äôs contents begin to be obscured by fog. Animatable.
Notes: A fog effect causes scene contents to become less visible the farther they are from the pointOfView node currently used for rendering. At distances less than the value of the fogStartDistance property, scene contents are fully visible. At greater distances, SceneKit blends the rendered scene contents with a constant color (specified by the fogColor property). At distances greater than the fogEndDistance property, the scene contents fade away completely and only the fog color is visible. Use fog to add atmospheric effects to your app or game, or to improve rendering performance by hiding parts of the scene that are far away from the current point of view.

The default start distance is 0.0.

You can animate changes to this property,Äôs value.
(Read and Write property)

13.47.20 lightingEnvironment as SCNMaterialPropertyMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A cube map texture that depicts the environment surrounding the scene,Äôs contents, used for advanced lighting effects.
Notes: When rendering materials with the physicallyBased lighting model, SceneKit illuminates surfaces differently according to the environment that surrounds them. For example, with physically based shading, even a diffuse surface takes on some color from the sky above it and the ground below it.

Tip: For realistic results, reuse the same contents for both the lighting environment and the background property.

For information about defining cube maps, see the discussion of the contents property.
(Read only property)

13.47.21 paused as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether to run actions, animations, particle systems, and physics simulations in the scene graph.
Notes: If false (the default), SceneKit continuously updates and renders the contents of the scene. Pausing

a scene pauses any running animations or actions attached to the scene graph, and suspends updates of the scene,Äôs physics simulation and any particle systems in the scene.
(Read and Write property)

13.47.22 physicsWorld as SCNPhysicsWorldMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The physics simulation associated with the scene.
Notes: Every scene automatically creates a physics world object to simulate physics on nodes in the scene. You use this property to access the scene,Äôs global physics properties, such as gravity, and to manage physics interactions between nodes. To make a node in the scene participate in the physics simulation, use either or both of its physicsBodyMBS and physicsFieldMBS properties.
(Read only property)

13.47.23 rootNode as SCNNodeMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The root node of the scene graph.
Notes: All scene content—nodes, geometries and their materials, lights, cameras, and related objects—is organized in a node hierarchy with a single common root node.

Some scene files created using external tools may describe node hierarchies containing multiple root nodes. When SceneKit imports such files, their separate root nodes will be made children of a new, unique root node.

Each child node,Äôs coordinate system is defined relative to the transformation of its parent node. You should not modify the transform property of the root node.
(Read only property)

13.47.24 attributeForKey(key as String) as Variant

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns or sets the scene attribute for the specified key.
Notes: key: One of the constants described in Scene Attributes that identifies the attribute to be written.
(Read and Write computed property)

13.48 class SCNShapeMBS

13.48.1 class SCNShapeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNShape represents a 2D shape (cubic Bezier spline) than can be extruded.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.48.2 Methods

13.48.3 Constructor(path as NSBezierPathMBS, extrusionDepth as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a 3D representation of the given shape with the given extrusion depth.
Notes: path: The cubic Bezier spline to extrude.
extrusionDepth: The extrusion depth.

13.48.4 shape(path as NSBezierPathMBS, extrusionDepth as double) as SCNShapeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a 3D representation of the given shape with the given extrusion depth.
Notes: path: The cubic Bezier spline to extrude.
extrusionDepth: The extrusion depth.

13.48.5 Properties

13.48.6 chamferMode as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The sides of the text that are chamfered.
Notes: The default value is ChamferModeBoth.
(Read and Write property)

13.48.7 chamferProfile as NSBezierPathMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Describes the profile used to when "chamferRadius" is not nil.
Notes: When "chamferProfile" is nil we fallback on a path representing a quadrant.
The profile should be a 2D curve beginning at (0,1) and ending at (1,0). The "flatness" property is also used to flatten this path. The default value is nil.
(Read and Write property)

13.48.8 chamferRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The chamfer radius. Animatable.
Notes: Values are clamped to the range [0, extrusionDepth / 2] . The default value is 0.
(Read and Write property)

13.48.9 extrusionDepth as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The extrusion depth. Animatable.
Notes: If the value is 0, we get a mono-sided, 2D version of the shape.
(Read and Write property)

13.48.10 path as NSBezierPathMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The path defining the shape to be rendered.
Notes: The path defines the outline of the shape. The path is filled using the even-odd rule. If the path is self-intersecting, the behavior is undefined.
(Read and Write property)

13.48.11 Constants

Chamfer Mode

Constant	Value	Description
ChamferModeBack	2	Apply a chamfer to only the back edge of the extruded shape.
ChamferModeBoth	0	Apply a chamfer to both front and back edges of the extruded shape.
ChamferModeFront	1	Apply a chamfer to only the front edge of the extruded shape.

13.49 class SCNSliderConstraintMBS

13.49.1 class SCNSliderConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNSliderConstraint constraint makes a node to collide and slide against a category of nodes
Notes: Subclass of the SCNConstraintMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.49.2 Methods

13.49.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNSliderConstraint object.

13.49.4 sliderConstraint as SCNSliderConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a SCNSliderConstraint object.

13.49.5 Properties

13.49.6 collisionCategoryBitMask as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the category of node to collide against. Defaults to 0.
Notes: (Read and Write property)

13.49.7 offset as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the offset of the slider.
Notes: Defaults to (0,0,0).

(Read and Write property)

13.49.8 Radius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Defines the radius of the slider. Defaults to 1.
Notes: (Read and Write property)

13.50 class SCNSphereMBS

13.50.1 class SCNSphereMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNSphere represents a sphere with controllable radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 21 to 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 35: On the Scene, Getting Started with SceneKit by Stefanie Juchmes
- 18.1, pages 32 to 33: On the Scene, Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 30: On the Scene, Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 27: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.50.2 Methods

13.50.3 Constructor(radius as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a sphere with given radius.
Notes: radius: The radius of the sphere.

13.50.4 sphere(radius as double) as SCNSphereMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a sphere with given radius.
Notes: radius: The radius of the sphere.

13.50.5 Properties

13.50.6 geodesic as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Indicate if the geometry is a geosphere.
Notes: The default value is false.
(Read and Write property)

13.50.7 radius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The sphere radius. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 0.5.
(Read and Write property)

13.50.8 segmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of segments along both spherical coordinates. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 48.
(Read and Write property)

13.51 class SCNTextMBS

13.51.1 class SCNTextMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNText represents a block of text that has been extruded
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

13.51.2 Methods

13.51.3 Constructor(text as string, extrusionDepth as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a 3D representation of given text with given extrusion depth.
Notes: string: The text to be represented.
extrusionDepth: The extrusion depth.

13.51.4 text(text as string, extrusionDepth as double) as SCNTextMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a 3D representation of given text with given extrusion depth.
Notes: string: The text to be represented.
extrusionDepth: The extrusion depth.
See also:

- 13.51.15 text as String 1141

13.51.5 Properties

13.51.6 alignmentMode as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines how individual lines of text are horizontally aligned within the bounds.
Notes: For the text to be aligned you first need to set its bounds, otherwise the text is not aligned. The default value is kCAAlignmentNatural.

(Read and Write property)

13.51.7 attributedString as NSAttributedStringMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The text to be represented.
Notes: The text can be string or NSAttributedStringMBS.
(Read and Write property)

13.51.8 chamferProfile as NSBezierPathMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Describes the profile used to when "chamferRadius" is not nil.
Notes: When "chamferProfile" is nil we fallback on a path representing a quadrant.
The profile should be a 2D curve beginning at (0,1) and ending at (1,0). The "flatness" property is also used to flatten this path. The default value is nil.
(Read and Write property)

13.51.9 chamferRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The chamfer radius. Animatable.
Notes: Values are clamped to the range [0, extrusionDepth / 2] . The actual chamfer radius might be different to the one here specified: large values are clipped to a per-glyph max value. The default value is 0.
(Read and Write property)

13.51.10 chamferSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop only.

Function: The number of chamfer subdivisions. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 10.
(Read and Write property)

13.51.11 containerFrame as CGRectMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A container within which the text may be wrapped or truncated.
Notes: The text will start at the top-left corner of the rect. You need to set this property for text truncation or alignment to work. Getting this property when it has never been set returns CGRectMBS with all values zero.
(Read and Write property)

13.51.12 extrusionDepth as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The extrusion depth. Animatable.
Notes: If the value is 0, we get a mono-sided, 2D version of the text.
(Read and Write property)

13.51.13 flatness as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the accuracy (or smoothness) with which fonts are rendered.
Notes: Smaller numbers give smoother curves at the expense of more computation and heavier geometries in terms of vertices. The default value is 0.6, which yields smooth curves.
(Read and Write property)

13.51.14 font as NSFontMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The font used to represent the text.
Notes: The font property is only used when the string property is not an NSAttributedString. Defaults to the system font (12 point).
(Read and Write property)

13.51.15 text as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The text to be represented.
Notes: The text can be string or NSAttributedStringMBS.

(Read and Write property)
See also:

- 13.51.4 text(text as string, extrusionDepth as double) as SCNTextMBS 1139

13.51.16 textSize as CGSizeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop only.

Function: Returns the bounding box size the receiver occupies.
Notes: (Read only property)

13.51.17 truncationMode as String

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Describes how the text is truncated to fit within the bounds.
Notes: For the text to be truncated you first need to set its bounds, otherwise the text is not truncated. The default value is kCATruncationNone.
(Read and Write property)

13.51.18 Wrapped as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines whether the text is wrapped to fit within the bounds.
Notes: For the text to be wrapped you first need to set its bounds, otherwise the text is not wrapped. The default value is false.
(Read and Write property)

13.52 class SCNTorusMBS

13.52.1 class SCNTorusMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNTorus represents a torus with controllable ring radius and pipe radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.2, page 37: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 30 to 31: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.52.2 Methods

13.52.3 Constructor(ringRadius as double, pipeRadius as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a torus with given ring radius and pipe radius.
Notes: ringRadius: The radius of the ring.
pipeRadius: The radius of the pipe.

13.52.4 torus(ringRadius as double, pipeRadius as double) as SCNTorusMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a torus with given ring radius and pipe radius.
Notes: ringRadius: The radius of the ring.
pipeRadius: The radius of the pipe.

13.52.5 Properties

13.52.6 pipeRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The radius of the torus pipe. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 0.25.
(Read and Write property)

13.52.7 pipeSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions of the pipe. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 24.
(Read and Write property)

13.52.8 ringRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The radius of the torus ring. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 0.5.
(Read and Write property)

13.52.9 ringSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions of the ring. Animatable.
Notes: If the value is less than 3, the behavior is undefined. The default value is 48.
(Read and Write property)

13.53 class SCNTransformConstraintMBS

13.53.1 class SCNTransformConstraintMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A SCNTransformConstraint applies on the transform of a node via a custom block.
Notes: Subclass of the SCNConstraintMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.53.2 Methods

13.53.3 Constructor

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

13.54 class SCNTubeMBS

13.54.1 class SCNTubeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: SCNTube represents a tube with controllable height, inner radius and outer radius.
Notes: Subclass of the SCNGeometryMBS class.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 18.4, page 44: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 31 to 33: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 22: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes

13.54.2 Methods

13.54.3 Constructor(InnerRadius as double, outerRadius as double, height as double)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a tube with given inner radius, outer radius and height.
Notes: innerRadius: The inner radius of the tube.
outerRadius: The outer radius of the tube.
height: The height of the tube.

13.54.4 tube(InnerRadius as double, outerRadius as double, height as double) as SCNTubeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates and returns a tube with given inner radius, outer radius and height.
Notes: innerRadius: The inner radius of the tube.
outerRadius: The outer radius of the tube.
height: The height of the tube.

13.54.5 Properties

13.54.6 height as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the tube. Animatable.
Notes: If the value is less than or equal to 0, the geometry is empty. The default value is 1.
(Read and Write property)

13.54.7 heightSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the Y axis. Animatable.
Notes: If the value is less than 1, the behavior is undefined. The default value is 1.
(Read and Write property)

13.54.8 innerRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The inner radius of the tube. Animatable.
Notes: If the value is less than or equal to 0, or if it is greater than or equal to the outer radius, then the geometry is empty. The default value is 0.25.
(Read and Write property)

13.54.9 outerRadius as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The outer radius of the tube. Animatable.
Notes: If the value is less than or equal to 0, or if it is less than or equal to the inner radius, then the geometry is empty. The default value is 0.5.
(Read and Write property)

13.54.10 radialSegmentCount as Integer

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The number of subdivisions along the radial coordinate. Animatable.

Notes: If the value is less than 3, the behavior is undefined. The default value is 48.
(Read and Write property)

13.55 class SCNVector3MBS

13.55.1 class SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A representation of a three-component vector.
Notes: SceneKit uses three-component vectors for a variety of purposes, such as describing node or vertex positions, surface normals, and scale or translation transforms. The different vector components should be interpreted based on the context in which the vector is being used.
Blog Entries

- Basic Forms in MBS Xojo SceneKit Plugin
- Physics example for SceneKit in Xojo
- MBS Xojo Plugins, version 19.1pr1

Xojo Developer Magazine

- 19.6, pages 47 to 48: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, pages 50 to 51: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.4, page 48: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.4, pages 43 to 44: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 36 to 38: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 34: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, pages 27 to 32: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.2, page 21: On the Scene Again (Part 2), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 37: On the Scene, Getting Started with SceneKit by Stefanie Juchmes
- 18.1, pages 31 to 33: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.55.2 Methods

13.55.3 Constructor(x as double = 0.0, y as double = 0.0, z as double = 0.0)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

13.55.4 copy as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the vector.

13.55.5 equals(other as SCNVector3MBS) as boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Compares two vectors.
Notes: Returns true if both are equal.

13.55.6 Null as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns vector with all values zero.

13.55.7 Vector(x as double, y as double, z as double) as SCNVector3MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new vector.

13.55.8 Properties

13.55.9 x as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The first component in the vector.
Notes: (Read and Write property)

13.55.10 y as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The second component in the vector.
Notes: (Read and Write property)

13.55.11 z as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The third component in the vector.
Notes: (Read and Write property)

13.56 class SCNVector4MBS

13.56.1 class SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A representation of a four-component vector.
Notes: SceneKit uses four-component vectors to represent multiple kinds of data:

- Axis-angle rotation or torque. The x, y, and z fields contain the normalized x-, y-, and z-components of the rotation axis, and the w field contains the rotation angle, in radians, or torque magnitude, in newton-meters.
- Color value (or range). The x, y, z, and w fields contain the red, green, blue, and alpha components of the color, or the width of the color variation range in each component.

Blog Entries

- MBS Xojo Plugins, version 19.1pr1

13.56.2 Methods

13.56.3 Constructor(x as double = 0.0, y as double = 0.0, z as double = 0.0, w as double = 0.0)

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.

13.56.4 copy as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the vector.

13.56.5 equals(other as SCNVector4MBS) as boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Compares two vectors.
Notes: Returns true if equal.

13.56.6 Null as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns vector with all values zero.

13.56.7 Vector(x as double, y as double, z as double, w as double) as SCNVector4MBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the vector with given values.

13.56.8 Properties

13.56.9 w as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The fourth component in the vector.
Notes: (Read and Write property)

13.56.10 x as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The first component in the vector.
Notes: (Read and Write property)

13.56.11 y as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The second component in the vector.
Notes: (Read and Write property)

13.56.12 z as Double

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The third component in the vector.
Notes: (Read and Write property)

13.57 class SCNViewMBS

13.57.1 class SCNViewMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A view for displaying 3D SceneKit content.
Notes: In macOS, SCNView is a subclass of NSView. As part of the operating system,Äôs view hierarchy, an SCNView object provides a place for SceneKit content in your app,Äôs user interface. You can create a SceneKit view by using its Constructor method. To provide content for a SceneKit view, assign an SCNSceneMBS object to its scene property.

For additional important methods and properties for working with SceneKit views, see the SCNSceneRenderer protocol. (You can also render SceneKit content into an arbitrary Metal command queue or OpenGL context using the SCNRenderer class, or into a Core Animation layer on macOS using the SCNLayer class. The SCNSceneRenderer protocol defines functionality common to all three SceneKit rendering classes.)

see
https://developer.apple.com/documentation/scenekit/scnview

Requires 64bit app on macOS 10.8 or newer
Subclass of the NSViewMBS class.
Blog Entries

- MBS Xojo Plugins, version 24.1pr4
- MBS Xojo Plugins, version 20.2pr1
- MBS Xojo Plugins, version 18.4pr8

Xojo Developer Magazine

- 19.6, page 46: Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes
- 18.4, page 39: On the Scene Again (Part 3), Getting Started with SceneKit by Stefanie Juchmes
- 18.1, page 26: On the Scene, Getting Started with SceneKit by Stefanie Juchmes

13.57.2 Methods

13.57.3 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new box view with size 100/100 and position 0/0
Example:

```
dim x as new SCNViewMBS
```

Notes: On success the handle property is not zero.

You can set PreferLowPowerDevice and PreferredRenderingAPI properties before initializing a view.
See also:

- 13.57.4 Constructor(Handle as Integer) 1156
- 13.57.5 Constructor(left as double, top as double, width as double, height as double) 1156

13.57.4 Constructor(Handle as Integer)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an object based on the given NSView handle.
Example:

```
dim t as new SCNViewMBS(0, 0, 100, 100)
dim v as new SCNViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a SCNView and the plugin retains this handle.
See also:

- 13.57.3 Constructor 1155
- 13.57.5 Constructor(left as double, top as double, width as double, height as double) 1156

13.57.5 Constructor(left as double, top as double, width as double, height as double)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new control with the given size and position.
Example:

```
dim left,top,width,height as Integer
// define rectangle
dim x as new SCNViewMBS(left, top, width, height)
```

Notes: On success the handle property is not zero.

You can set PreferLowPowerDevice and PreferredRenderingAPI properties before initializing a view.
See also:

- 13.57.3 Constructor 1155
- 13.57.4 Constructor(Handle as Integer) 1156

13.57.6 hitTest(Point as CGPointMBS, Options as Dictionary = nil) as SCNHitTestResultMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Searches the renderer,Äôs scene for objects corresponding to a point in the rendered image.
Notes: Point: A point in the screen-space (view, layer, or GPU viewport) coordinate system of the scene renderer.
options: A dictionary of options affecting the search. See Hit Testing Options Keys for acceptable values.

Returns an array of SCNHitTestResultMBS objects representing search results.

A 2D point in the rendered screen coordinate space can refer to any point along a line segment in the 3D scene coordinate space. Hit-testing is the process of finding elements of a scene located along this line segment. For example, you can use this method to find the geometry corresponding to a click event in a SceneKit view.

13.57.7 isNodeInsideFrustum(node as SCNNodeMBS, pointOfView as SCNNodeMBS) as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a Boolean value indicating whether a node might be visible from a specified point of view.
Notes: node: The node whose visibility is to be tested.
pointOfView: A node defining a point of view, as used by the pointOfView property.

Returns true if the bounding box of the tested node intersects the view frustum defined by the pointOfView node; otherwise, false.

Any node containing a camera or spotlight may serve as a point of view (see the pointOfView property for details). Such a node defines a viewing frustum—a portion of the scene,Äôs coordinate space, shaped like a

truncated pyramid, that encloses all points visible from that point of view.

Use this method to test whether a node lies within the viewing frustum defined by another node (which may or may not be the scene renderer,Äôs current pointOfView node). For example, in a game scene containing multiple camera nodes, you could use this method to determine which camera is currently best for viewing a moving player character.

Note that this method does not perform occlusion testing. That is, it returns true if the tested node lies within the specified viewing frustum regardless of whether that node,Äôs contents are obscured by other geometry.

13.57.8 nodesInsideFrustumWithPointOfView(pointOfView as SCNNodeMBS) as SCNNodeMBS()

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns all nodes that might be visible from a specified point of view.
Notes: pointOfView: A node defining a point of view, as used by the pointOfView property.

Returns an array of nodes whose bounding boxes intersect the view frustum defined by the pointOfView node. If the array is empty, no nodes lie within the specified frustum.

Any node containing a camera or spotlight may serve as a point of view (see the pointOfView property for details). Such a node defines a viewing frustum—a portion of the scene,Äôs coordinate space, shaped like a truncated pyramid, that encloses all points visible from that point of view.
Use this method find all nodes whose content lies within the viewing frustum defined by another node (which may or may not be the scene renderer,Äôs current pointOfView node).
Note that this method does not perform occlusion testing. That is, the returned array includes any node that lies within the specified viewing frustum regardless of whether that node,Äôs contents are obscured by other geometry.

13.57.9 pause

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Pauses playback of the view,Äôs scene.
Notes: This method has no effect if the scene is already paused.

13.57.10 play

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Resumes playback of the view,Äôs scene.
Notes: This method has no effect if the scene is not paused.

13.57.11 projectPoint(Point as SCNVector3MBS) as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Projects a point from the 3D world coordinate system of the scene to the 2D pixel coordinate system of the renderer.
Notes: point: A point in the world coordinate system of the renderer,Äôs scene.

Returns the corresponding point in the screen-space (view, layer, or GPU viewport) coordinate system of the scene renderer.

The z-coordinate of the returned point describes the depth of the projected point relative to the near and far clipping planes of the renderer,Äôs viewing frustum (defined by its pointOfView node). Projecting a point on the near clipping plane returns a point whose z-coordinate is 0.0; projecting a point on the far clipping plane returns a point whose z-coordinate is 1.0.

13.57.12 snapshot as NSImageMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Renders the view,Äôs scene into a new image object.

13.57.13 stop

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Stops playback of the view,Äôs scene and resets the scene time to its start time.

13.57.14 unprojectPoint(Point as SCNVector3MBS) as SCNVector3MBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Unprojects a point from the 2D pixel coordinate system of the renderer to the 3D world coordinate system of the scene.
Notes: point: A point in the screen-space (view, layer, or GPU viewport) coordinate system of the scene renderer.

Returns the corresponding point in the world coordinate system of the renderer,Äôs scene.

The z-coordinate of the point parameter describes the depth at which to unproject the point relative to the near and far clipping planes of the renderer,Äôs viewing frustum (defined by its pointOfView node). Unprojecting a point whose z-coordinate is 0.0 returns a point on the near clipping plane; unprojecting a point whose z-coordinate is 1.0 returns a point on the far clipping plane.

A 2D point in the rendered screen coordinate space can refer to any point along a line segment in the 3D scene coordinate space. To test for scene contents along this line—for example, to find the geometry corresponding to the location of a click event in a view—use the hitTest method.

13.57.15 Properties

13.57.16 allowsCameraControl as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the user can manipulate the current point of view that is used to render the scene.
Notes: If you set this property to true, SceneKit creates a camera node and handles mouse or touch events to allow the user to pan, zoom, and rotate their view of the scene. (Enabling user camera control does not modify camera objects already existing in the scene graph or the nodes containing them.)

When you enable user camera control, the defaultCameraController object handles input events and drives camera behavior. You can use that object's methods and properties to change the style of user camera interaction, and use the cameraControlConfiguration property to adjust control sensitivity.

In the default configuration, SceneKit provides the following controls:

- Pan with one finger to rotate the camera around the scene
- Pan with two fingers to translate the camera on its local xy-plane
- Pan with three fingers vertically to move the the camera forward backward
- Double-tap to switch to the next camera in the scene
- Rotate with two fingers to roll the camera (rotate on the camera node's z-axis)

Pinch to zoom in or zoom out (change the camera's fieldOfView)
The default value of this property is false. Use this option if you intend to control the camera programmatically.
(Read and Write property)

13.57.17 audioListener as SCNNodeMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The node representing the listener,Äôs position in the scene for use with positional audio effects.
Notes: When you use the SCNAudioPlayerMBS class to play sound, the resulting effect depends on the position of each audio source in the scene relative to the listener. For example, changes in relative position can cause a sound to be localized to the left or right channel for stereo headphone output.

This property determines the listener,Äôs position. If the value is nil (the default), the listener position is always the same as that of the pointOfView node. By providing a different node for this property, you can separate the listener position from the point of view—this produces an effect similar to that of a boom microphone in video production. For example, in a third-person game where the camera floats high in the sky above the player character, you might use the player character as the listener node so that sounds from positions nearest the player are loudest.
To place an audio source in the scene, use the addAudioPlayer method on an SCNNode object.
(Read and Write property)

13.57.18 autoenablesDefaultLighting as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit automatically adds lights to a scene.
Notes: If this property,Äôs value is false (the default), the only light sources SceneKit uses for rendering a scene are those contained in the scene graph. If you change the value to true, SceneKit automatically adds and places an omnidirectional light source when rendering scenes that contain no lights or only contain ambient lights.
(Read and Write property)

13.57.19 backgroundColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The background color of the view.
Notes: SceneKit displays this color behind the contents of the rendered scene. If the scene contents fill the view or if the scene provides its own background using the background property, the view,Äôs background color may not be visible.

This property,Äôs value must be a color that can be represented using RGBA components.
(Read and Write property)

13.57.20 debugOptions as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: Options for drawing overlay content in a scene that can aid debugging.
Notes: Use these options to display overlays that show otherwise-invisible scene content—such as node bounding boxes and the extents of physics fields—for use in debugging and profiling your app. For example:

- To visualize how well each object,Äôs physics representation corresponds to its visible geometry, show the shape of each SCNPhysicsBodyMBS object in the scene with the SCNDebugOptionShowPhysicsShapes option.
- To improve rendering performance in a scene with multiple SCNLightMBS objects, show each light,Äôs area of effect with the SCNDebugOptionShowLightExtents option and ensure that each object in the scene is affected by no more than three lights.

Debug options are bit mask patterns. To display multiple debugging overlays, combine options using the bitwise OR operator.
(Read and Write property)

13.57.21 defaultCameraController as SCNCameraControllerMBS

Plugin Version: 18.5, Platform: macOS, Targets: Desktop & iOS.

Function: The default camera controller for this view.
Notes: (Read only property)

13.57.22 jitteringEnabled as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit applies jittering to reduce aliasing artifacts.
Notes: Jittering is a process that SceneKit uses to improve the visual quality of a rendered scene. While the scene,Äôs content is still, SceneKit moves the pointOfView location very slightly (by less than a pixel in projected screen space). It then composites images rendered after several such moves to create the final rendered scene, creating an antialiasing effect that smooths the edges of rendered geometry.

By default, the value of this property is false, specifying that SceneKit should not perform jittering. Change the value to true to enable jittering.

Because the SCNViewMBS and SCNLayerMBS classes perform jittering automatically and asynchronously, enabling jittering for these classes has minimal impact on rendering performance. The SCNRenderer class performs jittering synchronously, incurring a high performance cost. With this class, jittering is suitable for rendering single frames on demand, but not for real-time rendering.
(Read and Write property)

13.57.23 loops as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit restarts the scene time after all animations in the scene have played.
Notes: If the value of this property is true (the default), SceneKit returns the scene time to zero after all animations associated with the scene have played, causing those animations to repeat. Otherwise, SceneKit stops playing the scene when all animations have completed.
(Read and Write property)

13.57.24 Playing as Boolean

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the scene is playing.
Notes: If the value of this property is false (the default), SceneKit does not increment the scene time, so animations associated with the scene do not play. Change this property,Äôs value to true to start animating the scene.
(Read and Write property)

13.57.25 pointOfView as SCNNodeMBS

Plugin Version: 19.1, Platform: macOS, Targets: Desktop & iOS.

Function: The node from which the scene,Äôs contents are viewed for rendering.
Notes: Use a node with an SCNCameraMBS instance assigned to its camera property to view a scene. The node provides the position and direction of a virtual camera, and the camera object provides rendering parameters such as field of view and focus.
For debugging lights and shadows, you can also designate a spotlight (an SCNLightMBS object whose type property is spot) as a point of view. In this case, the light,Äôs spotInnerAngle property determines the field of view, and its zNear and zFar properties determine the near and far extents of the region that is visible onscreen (also known as the viewing frustum).

In either case, the direction of view is along the negative z-axis of the node,Äôs local coordinate space.
(Read and Write property)

13.57.26 PreferLowPowerDevice as Integer

Plugin Version: 24.1, Platform: macOS, Targets: All.

Function: An option for whether to select low-power-usage devices for Metal rendering.
Notes: SceneKit uses this option when automatically selecting a Metal device on systems with multiple GPUs. If the value is 1, SceneKit uses a device with low power usage requirements—for example, the integrated GPU on a MacBook Pro with both integrated and discrete graphics hardware.
Leaving this key unspecified is equivalent to setting its value to 0. In this case, SceneKit chooses the most capable available Metal device.

Can be set before you create a new SCNViewMBS, SCNIOSControlMBS or SCNControlMBS.
Value is -1 if not set.
(Read and Write property)

13.57.27 preferredFramesPerSecond as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The animation frame rate that the view uses to render its scene.
Notes: SceneKit chooses an actual frame rate that is as close as possible to your preferred frame rate based on the capabilities of the screen the view is displayed on. The actual frame rate is usually a factor of the maximum refresh rate of the screen to provide a consistent frame rate. For example, if the maximum refresh rate of the screen is 60 frames per second, that is also the highest frame rate the view sets as the actual frame rate. However, if you ask for a lower frame rate, SceneKit might choose 30, 20, 15 or some other factor to be the actual frame rate. For this reason, you want to choose a frame rate that your app can consistently maintain.
The default value is 60 frames per second.
(Read and Write property)

13.57.28 PreferredRenderingAPI as Integer

Plugin Version: 24.1, Platform: macOS, Targets: All.

Function: The rendering API to use for rendering the view (for example, Metal or OpenGL).
Notes: The value for is one of the SCNRenderingAPI constants.

SceneKit attempts to initialize a view using the preferred API you specify in the SCNView initializer; if the current device does not support the preferred API, SceneKit automatically falls back to a supported API.

After initialization, use the renderingAPI property to find out whether a fallback occurred. For example, if you specify the SCNRenderingAPIMetal option when initializing a view on an iOS device that does not support Metal, SceneKit defaults to the SCNRenderingAPIOpenGLES2 option instead.

Can be set before you create a new SCNViewMBS, SCNIOSControlMBS or SCNControlMBS.
Value is -1 if not set.
(Read and Write property)

13.57.29 renderingAPI as Integer

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The graphics technology SceneKit uses to render the scene.
Notes: You choose a graphics technology when initializing a scene renderer:

- When initializing a SCNViewMBS object, use the Constructor initializer and the SCNPreferredRenderingAPIKey key. Alternatively, create a view in Interface Builder and use the Rendering API control in the inspector. During initialization, the view will attempt to use the preferred API, but will fall back to a different API if the preferred one is not supported on the current hardware.
- To create a SCNRendererMBS object that renders into your own OpenGL contect, use the rendererWithContext initializer. To create a renderer for use in your own Metal workflow, use the rendererWithDevice initializer.
- The rendering technology used by a SCNLayer object is determined by Core Animation.

After initializing a renderer, this property reflects the rendering technology in use.
(Read only property)

13.57.30 rendersContinuously as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether the view always renders at its preferred frame rate or only when its visible content changes.
Notes: When this value is false (the default), the view redraws its contents only when something in its scene graph change or animates. Use this option to maximize energy efficiency.

If you change this value to true, the view redraws itself continually, at the rate specified by the preferredFramesPerSecond property, regardless of whether content is changing or animating.
(Read and Write property)

13.57.31 scene as SCNSceneMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The scene to be displayed in the view.
Notes: (Read and Write property)

13.57.32 sceneTime as Double

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: The current scene time.
Notes: This timestamp determines how running animations behave, which is similar to how the playhead time in a video player application determines which frame of a movie to display. Scene time applies only to animations whose usesSceneTimeBase property is true, including those loaded from a scene source using the SCNSceneSourceAnimationImportPolicyPlayUsingSceneTimeBase option.

Use this property, together with the above animation options, when you want to directly control (or allow the user to directly control) the playback of animations. For example, if you,Äôre building an authoring tool for 3D assets, you might bind this property,Äôs value to a slider control for scrubbing through playback of animations in a scene file.
(Read and Write property)

13.57.33 showsStatistics as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether SceneKit displays rendering performance statistics in an accessory view.
Notes: The SceneKit statistics view displays various information about scene rendering performance and GPU resource usage, including a frames-per-second (fps) counter. In macOS, click the gear button in the statistics view to show a panel with additional controls for adjusting SceneKit,Äôs rendering of the scene.
(Read and Write property)

13.57.34 Constants

Antialiasing Modes

Constant	Value	Description
kAntialiasingModeMultisampling16X	4	Multisampling 16x
kAntialiasingModeMultisampling2X	1	Multisampling 2x
kAntialiasingModeMultisampling4X	2	Multisampling 4x
kAntialiasingModeMultisampling8X	3	Multisampling 8x
kAntialiasingModeNone	0	No antialiasing.

Debug Options

Constant	Value	Description
kDebugOptionNone	0	Disable all debugging overlays.
kDebugOptionRenderAsWireframe	64	Display only wireframe placeholders for geometries in the scene. Unlike the SCNDebugOptionShowWireframe option, this option disables n mal surface rendering, displaying only the wireframe for each geometry.
kDebugOptionShowBoundingBoxes	2	Display the bounding boxes for any nodes with content.
kDebugOptionShowCameras	1024	Display visualizations for nodes in the scene with attached cameras and th fields of view.
kDebugOptionShowConstraints	512	Display visualizations of the constraint objects acting on nodes in the scene
kDebugOptionShowCreases	256	Display nonsmoothed crease regions for geometries affected by surface sub vision.
kDebugOptionShowLightExtents	8	Display the regions affected by each SCNLightMBS object in the scene. Only lights whose type is SCNLightTypeOmni or SCNLightTypeSpot have area of effect; directional and ambient lights affect the entire scene.
kDebugOptionShowLightInfluences	4	Display the locations of each SCNLightMBS object in the scene.
kDebugOptionShowPhysicsFields	16	Display the regions affected by each SCNPhysicsFieldMBS object in the sce
kDebugOptionShowPhysicsShapes	1	Display the physics shapes for any nodes with attached SCNPhysicsBodyM objects.
kDebugOptionShowSkeletons	128	Display visualizations of the skeletal animation parameters for relevant geon tries.
kDebugOptionShowWireframe	32	Display geometries in the scene with wireframe rendering. When this option is enabled, SceneKit still renders scene geometry with associated materials, then overlays a wireframe rendering of the same geomet You can use this option, for example, to debug material rendering issues.

Rendering API

Constant	Value	Description
kRenderingAPIMetal	0	Use the Metal framework for SceneKit rendering. Metal provides improved graphics performance on supported devices, allows you to integrate GPU-compute tasks into a rendering workflow, and provides the same API in both iOS and macOS. Used by the renderingAPI property and the SCNPreferredRenderingAPIKey option when initializing an SCNViewMBS object.
kRenderingAPIOpenGLCore32	2	Use the OpenGL 3.2 Core Profile API for SceneKit rendering in macOS. Used by the renderingAPI property and the SCNPreferredRenderingAPIKey option when initializing an SCNViewMBS object.
kRenderingAPIOpenGLCore41	3	Use the OpenGL 4.1 Core Profile API for SceneKit rendering in macOS. Used by the renderingAPI property and the SCNPreferredRenderingAPIKey option when initializing an SCNViewMBS object.
kRenderingAPIOpenGLLegacy	1	Use the Legacy OpenGL API for SceneKit rendering in macOS. This option is available on all macOS systems supporting SceneKit. If you request the Metal rendering API for an SCNViewMBS object on a system that does not support Metal, SceneKit falls back to the Legacy OpenGL API. Used by the renderingAPI property and the SCNPreferredRenderingAPIKey option when initializing an SCNViewMBS object.

Chapter 14

SmartCard

14.1 class TKBERTLVRecordMBS

14.1.1 class TKBERTLVRecordMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An object that parses BER-encoded data and produces DER-encoded data for TLV records.
Notes: Available in macOS 10.12 or newer.
Subclass of the TKTLVRecordMBS class.

14.1.2 Methods

14.1.3 Constructor(tag as UInt64, data as MemoryBlock)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a BER-TLV record with the specified tag and value.
Notes: tag: The tag field of the record.
value: The value field of the record.

Returns a new TLV record containing the specified tag and value fields.
See also:

- 14.1.4 Constructor(tag as UInt64, records() as TKTLVRecordMBS) 1170

14.1.4 Constructor(tag as UInt64, records() as TKTLVRecordMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a BER-TLV record with the specified tag and an array of TLV subrecords.
Notes: tag: The tag field of the record.
records: The TLV subrecords of the record.

Returns a new TLV record containing the specified tag field and subrecords.
See also:

- 14.1.3 Constructor(tag as UInt64, data as MemoryBlock) 1169

14.1.5 dataForTag(tag as UInt64) as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Encodes a specified tag using BER-TLV tag encoding rules.
Notes: tag: The tag value to encode.

Returns a data object that encodes a tag value using BER-TLV encoding.

14.2 class TKCompactTLVRecordMBS

14.2.1 class TKCompactTLVRecordMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An object that implements encoding using Compact-TLV encoding according to ISO 7816-4.
Notes: Subclass of the TKTLVRecordMBS class.

14.2.2 Methods

14.2.3 Constructor(tag as integer, data as MemoryBlock)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a TLV record with the specified tag and value.
Notes: tag: The tag field of the record.
value: The value field of the record.

Returns a new TLV record containing the specified tag and value fields.

14.3 class TKSimpleTLVRecordMBS

14.3.1 class TKSimpleTLVRecordMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An object that implements encoding using Simple-TLV encoding according to ISO 7816-4.
Notes: Subclass of the TKTLVRecordMBS class.

14.3.2 Methods

14.3.3 Constructor(tag as integer, data as MemoryBlock)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a TLV record with the specified tag and value.
Notes: tag: The tag field of the record.
value: The value field of the record.

Returnes a new TLV record containing the specified tag and value fields.

14.4 class TKSmartCardATRInterfaceGroupMBS

14.4.1 class TKSmartCardATRInterfaceGroupMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A single interface-bytes group for a Smart Card ATR (Answer to Reset).
Notes: You access instances of this class by calling the interfaceGroupAtIndex and interfaceGroupForProtocol methods on an TKSmartCardATRMBS object.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

14.4.2 Methods

14.4.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

14.4.4 Properties

14.4.5 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.4.6 HasProtocol as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether protocol is defined.
Notes: (Read only property)

14.4.7 HasTA as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether we have a TA value defined.
Notes: (Read only property)

14.4.8 HasTB as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether we have a TB value defined.
Notes: (Read only property)

14.4.9 HasTC as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether we have a TC value defined.
Notes: (Read only property)

14.4.10 Protocol as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The protocol for this group.
Notes: This property returns the protocol number. See kProtocol* constants.
This property is zero for the first interface group (global), as it has no assigned protocol.
(Read only property)

14.4.11 TA as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The TA interface byte of ATR group.
Notes: (Read only property)

14.4.12 TB as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The TB interface byte of ATR group.
Notes: (Read only property)

14.4.13 TC as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The TC interface byte of ATR group.
Notes: (Read only property)

14.5 class TKSmartCardATRMBS

14.5.1 class TKSmartCardATRMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A parsed ATR (Answer To Reset) message from a Smart Card.
Notes: This class declares a programmatic interface to parsing an ATR from data or a byte stream, and accessing the individual parts.

The TKSmartCardATR class parses ATR messages according to the ISO/IEC 7816-3 specification.

14.5.2 Methods

14.5.3 available as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on MacOS 10.10 or newer.

14.5.4 Constructor(data as MemoryBlock)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a TKSmartCardATR object from a provided data object.
Notes: bytes: The ATR data to be parsed.

Returns a TKSmartCardATR object initialized with the parsed data. If bytes does not contain a valid ATR, raises UnsupportedOperationException.

14.5.5 historicalRecords as TKCompactTLVRecordMBS()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A list of compact TLV records parsed from historical bytes.

14.5.6 interfaceGroupAtIndex(index as Integer) as TKSmartCardATRInterfaceGroupMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the interface group at the specified index.
Notes: index: The index of the desired interface group.

Interface group indexes start at 1, as specified by ISO 7816-3.

Returns the interface group at the specified index, or nil if not present.

14.5.7 interfaceGroupForProtocol(protocol as Integer) as TKSmartCardATRInterfaceGroupMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the interface group with the specified protocol.
Notes: protocol: The protocol used by the desired interface group.

Returns the interface group with the specified protocol, or nil if none exists.

14.5.8 protocols as Integer()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An array of protocols indicated in the ATR.
Notes: Each element in the returned array is an integer value corresponding to a member of the kProtocol* constants.
The returned protocols are ordered such that the default protocol is at index 0, and any duplicate values are removed.

14.5.9 Properties

14.5.10 bytes as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The ATR message data.
Notes: (Read only property)

14.5.11 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.5.12 historicalBytes as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The ATR historical bytes, not including interface bytes or the TCK (check byte).
Notes: (Read only property)

14.5.13 Constants

Protocols

Constant	Value	Description
kProtocolAny	65535	Any available transmission protocols.
kProtocolNone	0	No transmission protocols.
kProtocolT0	1	T=0 transmission protocol.
kProtocolT1	2	T=1 transmission protocol.
kProtocolT15	32768	T=15 transmission protocol.

14.6 class TKSmartCardMBS

14.6.1 class TKSmartCardMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of a smart card.
Notes: This class provides an interface for managing sessions with a smart card, transmitting requests, and facilitating user interaction.
You can create a TKSmartCardMBS object when a smart card is inserted into a slot, by calling the makeSmartCard method on the corresponding TKSmartCardSlotMBS object. To start communicating with the smart card, call the beginSession method on the TKSmartCardMBS object. Once an exclusive session has been established, you transmit data using the transmitRequest:reply: method. After you,Äôve finished communicating with a smart card, you call the endSession method.
If the smart card is physically removed from its slot, the session object becomes invalid, and any further calls to transmitRequest will return an error.

14.6.2 Methods

14.6.3 beginSession(tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Begins a session with the Smart Card.
Notes: This method will fail if there is already an existing session for the Smart Card.
Calls to this method must be balanced with calls to endSession.

Calls beginSessionCompleted event later.

14.6.4 beginSessionSync(byref error as NSErrorMBS) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Begins a session with the Smart Card.
Notes: This is the synchronous version, which waits for callback from operation system.
Returns true on success and false on failure.

This method will fail if there is already an existing session for the Smart Card.
Calls to this method must be balanced with calls to endSession.

14.6.5 beginSessionWithDelegate(handler as beginSessionCompletedDelegateMBS, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Transmits data in Application Protocol Data Unit (APDU) format to the Smart Card.
Notes: request: The APDU request data.

You should only call this method after a session to the Smart Card has been established using the beginSessionWithReply: method, and before the session is terminated using the endSession method.

Invokes handler later, which has this declaration:
beginSessionCompletedDelegate(success as Boolean, error as NSErrorMBS, tag as Variant)

14.6.6 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.6.7 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.6.8 endSession

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Completes any pending transmissions and ends the session to the Smart Card.
Notes: Calls to this method should balance calls to beginSession.

14.6.9 inSession(byref error as NSErrorMBS, tag as variant = nil) as boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Synchronously begins a session, executes the given block, and ends the session.
Notes: error: On return, if an error occurred when attempting to create a session or execute the block,

contains details about the error.

Returns true if the session was successfully created and the inSession event returns true; otherwise, false.
See also:

- 14.6.32 inSession(byref error as NSErrorMBS, tag as Variant) as Boolean 1187

14.6.10 readFileWithDelegate(FileName as MemoryBlock, handler as readFileCompletedDelegateMBS, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Reads file with given filename.
Notes: This is for Belgian ID card to read a file.

You should only call this method after a session to the Smart Card has been established using the beginSessionWithReply: method, and before the session is terminated using the endSession method.

Invokes handler later, which has this declaration:
readFileCompletedDelegateMBS(FileName as MemoryBlock, Content as MemoryBlock, error as NSErrorMBS, tag as Variant)

14.6.11 sendIns(ins as UInt8, p1 as UInt8, p2 as UInt8, requestData as MemoryBlock, le as Integer = 0, byref sw as Uint16, byref error as NSErrorMBS) as Memoryblock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Synchronously transmits an APDU command to the card and returns the response.
Notes: ins: The instruction code.
p1: The first parameter.
p2: The second parameter.
requestData: The data field of the APDU, or nil if no input data field should be present—for example, a case1 or case2 APDU.
The length of the data serves as Lc field of the APDU.
le: The expected number of bytes to be returned, or zero if no output data are expected—for example, a case1 or case3 APDU. Pass 0 to accept as many bytes as the card provides.
sw: On return, contains the result code as represented by the first two bytes (SW1SW2) of the returned data.
error: On return, if an error occurred when attempting to create a session or execute the block, contains details about the error.

Returns the returned data without the first two bytes (SW1SW2), or nil if an error occurred.
See also:

- 14.6.12 sendIns(ins as UInt8, p1 as UInt8, p2 as UInt8, requestData as MemoryBlock, le as Integer = 0, tag as variant = nil) 1182

14.6.12 sendIns(ins as UInt8, p1 as UInt8, p2 as UInt8, requestData as MemoryBlock, le as Integer = 0, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Asynchronously transmits an APDU command to the card, returning the response in a completion handler.
Notes: ins: The instruction code.
p1: The first parameter.
p2: The second parameter.
requestData: The data field of the APDU, or nil if no input data field should be present—for example, a case1 or case2 APDU.
The length of the data serves as Lc field of the APDU.
le: The expected number of bytes to be returned, or zero if no output data are expected—for example, a case1 or case3 APDU. Pass 0 to accept as many bytes as the card provides.

Calls sendInsCompleted event later.
See also:

- 14.6.11 sendIns(ins as UInt8, p1 as UInt8, p2 as UInt8, requestData as MemoryBlock, le as Integer = 0, byref sw as Uint16, byref error as NSErrorMBS) as Memoryblock 1181

14.6.13 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the observer to call validChanged event.
Notes: Called for you by constructor.

14.6.14 transmitRequest(request as MemoryBlock, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Transmits data in Application Protocol Data Unit (APDU) format to the Smart Card.
Notes: request: The APDU request data.

You should only call this method after a session to the Smart Card has been established using the beginSessionWithReply: method, and before the session is terminated using the endSession method.

Calls transmitRequestCompleted method later.

14.6.15 transmitRequestSync(request as MemoryBlock, byref response as MemoryBlock, byref error as NSErrorMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Transmits data in Application Protocol Data Unit (APDU) format to the Smart Card.
Notes: This is the synchronous version, which waits for callback from operation system.

request: The APDU request data.
response: The response.

You should only call this method after a session to the Smart Card has been established using the beginSessionWithReply: method, and before the session is terminated using the endSession method.

14.6.16 transmitRequestWithDelegate(request as MemoryBlock, handler as transmitRequestCompletedDelegateMBS, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Transmits data in Application Protocol Data Unit (APDU) format to the Smart Card.
Notes: request: The APDU request data.

You should only call this method after a session to the Smart Card has been established using the beginSessionWithReply: method, and before the session is terminated using the endSession method.

Invokes handler later, which has this declaration:
transmitRequestCompletedDelegate(request as MemoryBlock, response as MemoryBlock, error as NSErrorMBS, tag as Variant)

14.6.17 userInteractionForSecurePINChange(PINFormat as TKSmartCardPINFormatMBS, APDU as MemoryBlock, currentPINByteOffset as Integer, newPINByteOffset as Integer) as TKSmartCardUserInteractionForSecurePINChangeMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a new user interaction object for secure PIN change using the smart card reader facilities (typically a HW keypad).
Notes: PINFormat: The PIN format descriptor.
APDU: The Application Protocol Data Unit (APDU) used by the Smart Card to fill in PIN data.
currentPINByteOffset: The offset, in bytes, within the Application Protocol Data Unit (APDU) field to mark a location of a PIN block for filling in the entered PIN.
newPINByteOffset: The offset, in bytes, within the Application Protocol Data Unit (APDU) field to mark a location of a PIN block for filling in the new PIN.

Returns a new user interaction object for secure PIN verification, or nil if this feature is not supported by the Smart Card reader.

You should only call this method after a session to the Smart Card has been established using the beginSession method, and before the session is terminated using the endSession method.
Once the interaction has been successfully completed, the results are available via the resultData and resultSW properties of the returned TKSmartCardUserInteractionForSecurePINVerificationMBS object.

14.6.18 userInteractionForSecurePINVerification(PINFormat as TKSmartCardPINFormatMBS, APDU as MemoryBlock, PINByteOffset as Integer) as TKSmartCardUserInteractionForSecurePINVerificationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates and returns a new user interaction object for secure PIN verification using the Smart Card reader facilities.
Notes: PINFormat: The PIN format descriptor.
APDU: The Application Protocol Data Unit (APDU) used by the Smart Card to fill in PIN data.
PINByteOffset: The offset, in bytes, within the Application Protocol Data Unit (APDU) field to mark a location of a PIN block for filling in the entered PIN.

This parameter is not currently used. Pass 0.

Returns a new user interaction object for secure PIN verification, or nil if this feature is not supported by the Smart Card reader.

You should only call this method after a session to the Smart Card has been established using the beginSession method, and before the session is terminated using the endSession method.
Once the interaction has been successfully completed, the results are available via the resultData and resultSW properties of the returned TKSmartCardUserInteractionForSecurePINVerificationMBS object.

14.6.19 Properties

14.6.20 AllowedProtocols as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The protocols allowed for communication with the Smart Card.
Notes: kProtocolAny by default.
This property is consulted only when beginning a session to a Smart Card. Any changes to this property will not be reflected by the current session, if one is already established.
(Read and Write property)

14.6.21 cla as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The CLA byte used for APDU transmission.
Notes: 0 by default.
(Read and Write property)

14.6.22 Context as Variant

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: User-specified information.
Notes: This property is automatically set to nil if the Smart Card is removed or another TKSmartCardMBS object begins a session.
(Read and Write property)

14.6.23 CurrentProtocol as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The protocol used for communication with the Smart Card.
Notes: Returns kProtocolNone if no session is currently established.
(Read only property)

14.6.24 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.6.25 Sensitive as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether sessions established for the Smart Card should be considered sensitive.
Notes: False by default.
When this property is set to true, any sessions established for the receiver will begin and end by sending a reset command to the Smart Card. This is recommended anytime potentially sensitive information is transferred.
(Read and Write property)

14.6.26 Slot as TKSmartCardSlotMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The slot in which the Smart Card is inserted.
Notes: (Read only property)

14.6.27 UseCommandChaining as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether to use command chaining of APDU with a data field longer than 255 bytes.
Notes: By default, this property is set to true when the Smart Card ATR announces that command chaining is supported.
(Read and Write property)

14.6.28 UseExtendedLength as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether to use extended length APDU.
Notes: By default, this property is set to true when the Smart Card slot supports transmitting extended length commands, and the ATR announces that extended length APDU is supported.
(Read and Write property)

14.6.29 Valid as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the Smart Card is valid and accessible from its slot.
Notes: (Read only property)

14.6.30 Events

14.6.31 beginSessionCompleted(success as Boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when session begun.
Notes: success: Whether the session could be established successfully.
error: Contains information about the the error preventing the transaction from being established.
The NSError object is created in the TKErrorDomain domain with a code in the TKErrorCode enumeration.

14.6.32 inSession(byref error as NSErrorMBS, tag as Variant) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: The event called in session.
Notes: The event to be called in the context of the created session. This event returns a Boolean value indicating whether communication was successful.

error: To indicate that an error occurred during the session, populate this with an NSError object containing details and return false.
See also:

- 14.6.9 inSession(byref error as NSErrorMBS, tag as variant = nil) as boolean 1180

14.6.33 sendInsCompleted(ins as UInt8, p1 as UInt8, p2 as UInt8, requestData as MemoryBlock, le as integer, replyData as MemoryBlock, sw as UInt16, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when sendIns completed.
Notes: replyData: The returned data without the first two bytes (SW1SW2), or nil if an error occurred.
sw: The result code as represented by the first two bytes (SW1SW2) of the returned data.
error: If a communication error occurred or the sw result code is anything other than &h9000, contains

details about the error.

14.6.34 transmitRequestCompleted(request as MemoryBlock, response as MemoryBlock, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: The event called when transmit is done.
Notes: response: The APDU response data, or nil if communication with the Smart Card failed.
error: Contains information about the the error preventing the transaction from being established.
The NSError object is created in the TKErrorDomain domain with a code in the TKErrorCode enumeration.

14.6.35 ValidChanged

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when valid property changed.

14.6.36 Constants

Protocols

Constant	Value	Description
kProtocolAny	65535	Any available transmission protocols.
kProtocolNone	0	No transmission protocols.
kProtocolT0	1	T=0 transmission protocol.
kProtocolT1	2	T=1 transmission protocol.
kProtocolT15	32768	T=15 transmission protocol.

Errors

Constant	Value	Description
TKErrorCodeAuthenticationFailed	-5	Authentication failed.
TKErrorCodeAuthenticationNeeded	-9	Authentication is needed.
TKErrorCodeBadParameter	-8	An invalid parameter was provided.
TKErrorCodeCanceledByUser	-4	The operation was canceled by the user.
TKErrorCodeCommunicationError	-2	A communication error occurred.
TKErrorCodeCorruptedData	-3	The data was corrupted.
TKErrorCodeNotImplemented	-1	The functionality is not implemented.
TKErrorCodeObjectNotFound	-6	The object was not found.
TKErrorCodeTokenNotFound	-7	The token was not found.

14.6.37 Delegates

14.6.38 beginSessionCompletedDelegateMBS(success as Boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The delegate for the beginSessionWithDelegate method.

14.6.39 readFileCompletedDelegateMBS(FileName as MemoryBlock, Content as MemoryBlock, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The delegate for the readFileWithDelegate method.

14.6.40 transmitRequestCompletedDelegateMBS(request as MemoryBlock, response as MemoryBlock, error as NSErrorMBS, tag as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The delegate for the transmitRequestWithDelegate method.

14.7 class TKSmartCardPINFormatMBS

14.7.1 class TKSmartCardPINFormatMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The formatting properties for a PIN, such as character encoding and length constraints.
Notes: You typically interact with TKSmartCardPINFormatMBS objects when calling the userInteractionForSecurePINChange and userInteractionForSecurePINVerification methods on an instance of TKSmartCardMBS.

Available in macOS 10.11 or newer.

14.7.2 Methods

14.7.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.7.4 Properties

14.7.5 Charset as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The format of PIN characters.
Notes: kCharsetNumeric by default.
(Read and Write property)

14.7.6 Encoding as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The encoding of PIN characters.
Notes: kEncodingASCII by default.
(Read and Write property)

14.7.7 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.7.8 MaxPINLength as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The maximum number of characters to form a valid PIN.
Notes: 8 by default.
(Read and Write property)

14.7.9 MinPINLength as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The minimum number of characters to form a valid PIN.
Notes: 4 by default.
(Read and Write property)

14.7.10 PINBitOffset as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The offset, in bits, within the PIN block to mark a location for filling in the formatted PIN, which is justified with respect to the PINJustification property value.
Notes: 0 by default.

The value of PINBitOffset indirectly controls the internal system units indicator. If PINBitOffset is byte aligned (that is, PINBitOffset % 8 == 0), the internal representation of PINBitOffset gets converted from bits to bytes.
(Read and Write property)

14.7.11 PINBlockByteLength as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The total length of the PIN block in bytes.
Notes: 8 by default.
(Read and Write property)

14.7.12 PINJustification as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The justification within the PIN block.
Notes: kJustificationLeft by default.
(Read and Write property)

14.7.13 PINLengthBitOffset as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The offset, in bits, within the PIN block to mark a location for filling in the PIN length, which is always left justified.
Notes: 0 by default.

The value of PINLengthBitOffset indirectly controls the internal system units indicator. If PINLengthBitOffset is byte aligned (that is, PINLengthBitOffset % 8 == 0), the internal representation of PINLengthBitOffset gets converted from bits to bytes.
(Read and Write property)

14.7.14 PINLengthBitSize as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The size, in bits, of the PIN length field. If set to 0, PIN length is not written.
Notes: 0 by default.
(Read and Write property)

14.7.15 Constants

PIN character sets

Constant	Value	Description
kCharsetAlphanumeric	1	PIN can be composed of digits and letters.
kCharsetNumeric	0	PIN is only composed of digits.
kCharsetUpperAlphanumeric	2	PIN can be composed of digits and uppercase letters.

Encodings

Constant	Value	Description
kEncodingASCII	1	Characters are encoded in ASCII format (for example, 1234 is encoded as 31h 32h 33h 34h).
kEncodingBCD	2	Characters (only digits) are encoded in BCD format (for example, 1234 is encoded as 12h 34h).
kEncodingBinary	0	Characters are encoded in Binary format (for example, 1234 is encoded as 01h 02h 03h 04h).

PIN Justification

Constant	Value	Description
kJustificationLeft	0	Justify to the left.
kJustificationRight	1	Justify to the right.

14.8 class TKSmartCardSlotManagerMBS

14.8.1 class TKSmartCardSlotManagerMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An interface to all available smart card reader slots.
Notes: Get a list of all known smart card reader slots in the system using the slotNames property, and access individual slots by name using the getSlotWithName method.

The com.apple.security.smartcard entitlement is required in order to use TKSmartCardSlotManager.

14.8.2 Methods

14.8.3 available as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on MacOS 10.11 or newer.

14.8.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.8.5 defaultManager as TKSmartCardSlotManagerMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The shared singleton Smart Card reader slot manager.
Notes: This method returns nil unless the com.apple.security.smartcard entitlement is enabled.

14.8.6 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.8.7 getSlotWithName(name as string, tag as variant = nil)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Asynchronously calls a block with a Smart Card reader slot for a specified name.
Notes: name: The name of the Smart Card reader slot.

Calls gotSlotWithName later.

14.8.8 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the observer to call slowNamesChanged event.
Notes: Called for you by constructor.

14.8.9 slotNamed(name as string) as TKSmartCardSlotMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the Smart Card slot with a given name.
Notes: Returns the slot with the specified name, or nil if no slot with that name exists.

14.8.10 slotNames as string()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A list of identifiers for all the Smart Card reader slots available to the system.

14.8.11 Properties

14.8.12 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.8.13 Events

14.8.14 gotSlotWithName(name as string, slot as TKSmartCardSlotMBS, tag as variant)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: The event called when the slow with the name is found.

14.8.15 slotNamesChanged

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when slot name list changed.

14.9 class TKSmartCardSlotMBS

14.9.1 class TKSmartCardSlotMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A single smart card reader slot in the system.
Notes: Use the TKSmartCardSlotManagerMBS class to manage all the smart card reader slots available to the system. You can retrieve the names of available smart card reader slots for a system using the slotNames property of a manager object, and access instances of TKSmartCardSlotMBS using the getSlotWithName method.

14.9.2 Methods

14.9.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.9.4 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.9.5 makeSmartCard as TKSmartCardMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a new TKSmartCardMBS object representing the currently inserted Smart Card.
Notes: A new TKSmartCardMBS object, or nil if no Smart Card is currently inserted.

You can create multiple instances of TKSmartCardMBS that represent the same Smart Card. Exclusivity of data transfer is handled by sessions on the individual TKSmartCardMBS objects.

14.9.6 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the observer to call stateChanged event.
Notes: Called for you by constructor.

14.9.7 Properties

14.9.8 ATR as TKSmartCardATRMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The ATR (Answer to Reset) of the inserted Smart Card, or nil if no Smart Card is inserted or the inserted Smart Card is mute.
Notes: (Read only property)

14.9.9 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.9.10 maxInputLength as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The maximum length of input APDU (Application Protocol Data Unit) that the Smart Card reader slot is able to transfer to the Smart Card.
Notes: (Read only property)

14.9.11 maxOutputLength as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The maximum length of output APDU (Application Protocol Data Unit) that the Smart Card reader slot is able to transfer from the Smart Card.
Notes: (Read only property)

14.9.12 Name as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The name of the Smart Card reader slot.
Notes: (Read only property)

14.9.13 State as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The current state of the Smart Card reader slot.
Notes: (Read only property)

14.9.14 Events

14.9.15 StateChanged

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when state changed.

14.9.16 Constants

States

Constant	Value	Description
kStateEmpty	1	The Smart Card reader slot is empty; no card is inserted.
kStateMissing	0	The Smart Card reader slot is no longer known to the system. This is the terminal state of a TKSmartCardSlotThis instance; once it has reached this state, the Smart Card reader slot cannot be reinitialized.
kStateMuteCard	3	A Smart Card is inserted, but is mute, or does not provide responses to commands.
kStateProbing	2	A Smart Card was inserted into the slot and an initial probe is in underway.
kStateValidCard	4	A Smart Card is inserted and properly answered to a reset command.

14.10 class TKSmartCardTokenDriverMBS

14.10.1 class TKSmartCardTokenDriverMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The driver that acts as an entry point for smart card app extensions.
Notes: Subclass of the TKTokenDriverMBS class.

14.10.2 Methods

14.10.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.11 class TKSmartCardTokenMBS

14.11.1 class TKSmartCardTokenMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of a smart card based cryptographic token.#
Notes: Subclass of the TKTokenMBS class.

14.11.2 Methods

14.11.3 Constructor(smartCard as TKSmartCardMBS, AID as MemoryBlock, instanceID as String, tokenDriver as TKSmartCardTokenDriverMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a smart card token with the specified smart card, application identifier, and token driver.
Notes: smartCard: The smart card on which the created token should operate.
AID: The ISO 7816-4 application identifier for the smart card.
instanceID: A unique, persistent identifier for this token. This value is typically generated from the serial number of the target hardware.
tokenDriver: The driver associated with the created token.
See also:

- 14.11.4 Constructor(tokenDriver as TKTokenDriverMBS, instanceID as String) 1201

14.11.4 Constructor(tokenDriver as TKTokenDriverMBS, instanceID as String)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.11.3 Constructor(smartCard as TKSmartCardMBS, AID as MemoryBlock, instanceID as String, tokenDriver as TKSmartCardTokenDriverMBS) 1201

14.11.5 Properties

14.11.6 AID as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The ISO 7816-4 application identifiers of the Smart Card.
Notes: This value is specified in the Smart Card token extension's NSExtensionAttributes property list by the com.apple.ctk.aid attribute. If this attribute specifies multiple AIDs, this parameter represents the application identifier found on the card that is already preselected. If the com.apple.ctk.aid attribute is not present, no application is automatically preselected and the value of this property is nil.
(Read only property)

14.12 class TKSmartCardTokenSessionMBS

14.12.1 class TKSmartCardTokenSessionMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A token session that is based on a smart card token.
Notes: You can use the smartCard property to access and send APDUs to the underlying smart card.

Available on MacOS 10.12 or newer.
Subclass of the TKTokenSessionMBS class.

14.12.2 Methods

14.12.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.12.4 Constructor(token as TKTokenMBS) 1203

14.12.4 Constructor(token as TKTokenMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token session with the specified token.
Notes: token: The token to which the initialized session is bound.

Returns a new token session created with the specified token.
See also:

- 14.12.3 Constructor 1203

14.12.5 Properties

14.12.6 smartCard as TKSmartCardMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The smart card for the active exclusive session and selected application.
Notes: This property can only be accessed in the implementation of a TKTokenSessionDelegate protocol

delegate method. If the associated token has a value set for the AID property, this property opens an exclusive session to the card, with the application already selected.

You should not call beginSession or endSession on the returned value. Instead, the system will take care of beginning the exclusive session and terminating it when the current token request servicing is finished.
You can store any kind of information representing state of the card using the context property. This property will be automatically set to nil if the card is reset or accessed by different TKSmartCard instance, such as by another process. You can check the context property for any previously stored values as a way to avoid costly state restoration before performing an operation.
(Read only property)

14.13 class TKSmartCardUserInteractionForPINOperationMBS

14.13.1 class TKSmartCardUserInteractionForPINOperationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of user interaction for secure PIN operations on a Smart Card reader.
Notes: There are two types of user interactions: those for secure PIN change and those for secure PIN validation. These interactions are instances of the TKSmartCardUserInteractionForSecurePINChangeMBS, or TKSmartCardUserInteractionForSecurePINVerificationMBS subclasses of TKSmartCardUserInteraction-ForPINOperationMBS, respectively.

You interact with instances of one of the subclasses of TKSmartCardUserInteractionForPINOperationMBS when calling the userInteractionForSecurePINChange and userInteractionForSecurePINVerification methods on an TKSmartCard object.
The result of a user interaction is available once the interaction has completed.

Available in macOS 10.11 or newer.
Subclass of the TKSmartCardUserInteractionMBS class.

14.13.2 Methods

14.13.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.13.4 PINMessageIndices as Integer()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A list of message indices referring to a predefined message table, used to specify the type and number of messages displayed during the PIN operation. nil by default.
Notes: If nil, the reader does not display any message (reader specific). Typically, PIN verification takes 1 message; PIN modification takes 1 –3 messages.

14.13.5 Properties

14.13.6 locale as NSLocaleMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The locale for the displayed messages.
Notes: If nil, the user,Äôs current locale is used. By default, this value is the current locale of the system. (Read and Write property)

14.13.7 PINCompletion as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The conditions under which PIN entry should be considered complete.
Notes: (Read and Write property)

14.13.8 resultData as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The returned data without SW1-SW2 bytes, if any.
Notes: (Read and Write property)

14.13.9 resultSW as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The SW1-SW2 status bytes.
Notes: (Read and Write property)

14.13.10 Constants

PIN Completion

Constant	Value	Description
kPINCompletionKey	2	Key
kPINCompletionMaxLength	1	MaxLength
kPINCompletionTimeout	4	Timeout

14.14 class TKSmartCardUserInteractionForSecurePINChangeMBS

14.14.1 class TKSmartCardUserInteractionForSecurePINChangeMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of the user interaction for secure PIN change operations on a Smart Card reader.
Notes: The result of a user interaction is available once the interaction has completed.
Available in macOS 10.11+.
Subclass of the TKSmartCardUserInteractionForPINOperationMBS class.

14.14.2 Methods

14.14.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.14.4 Properties

14.14.5 PINConfirmation as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The way PIN confirmation is requested.
Notes: kPINConfirmationNone by default.
(Read and Write property)

14.14.6 Constants

Pin Confirmation

Constant	Value	Description
kPINConfirmationCurrent	2	Current PIN confirmation.
kPINConfirmationNew	1	New PIN confirmation.
kPINConfirmationNone	0	No PIN confirmation.

14.15 class TKSmartCardUserInteractionForSecurePINVerification-MBS

14.15.1 class TKSmartCardUserInteractionForSecurePINVerificationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of the user interaction for secure PIN change verification on a Smart Card reader.
Notes: The result of a user interaction is available once the interaction has completed.
Subclass of the TKSmartCardUserInteractionForPINOperationMBS class.

14.15.2 Methods

14.15.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.16 class TKSmartCardUserInteractionMBS

14.16.1 class TKSmartCardUserInteractionMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The base class for encapsulating user interaction with a Smart Card reader.
Notes: There are two types of user interactions: those for secure PIN change and those for secure PIN validation. These interactions are instances of the TKSmartCardUserInteractionForSecurePINChangeMBS, or TKSmartCardUserInteractionForSecurePINVerificationMBS subclasses of TKSmartCardUserInteraction-ForPINOperationMBS, respectively. TKSmartCardUserInteractionForPINOperationMBS is a subclass of TKSmartCardUserInteractionMBS.

You interact with instances of one of the subclasses of TKSmartCardUserInteractionForPINOperationwhen-MBS calling the userInteractionForSecurePINChange and userInteractionForSecurePINVerification methods on an TKSmartCardMBS object.

Available in macOS 10.11+.

14.16.2 Methods

14.16.3 Cancel as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Attempts to cancel an interaction started by calling run.
Notes: Returns false if the operation is not running, or if cancelation is not supported.
For certain interactions, cancellation may not be available.

14.16.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.16.5 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.16.6 Run

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Runs the user interaction and asynchronously receives a reply.
Notes: Calls RunCompleted event later.

14.16.7 Properties

14.16.8 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.16.9 initialTimeout as Double

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The timeout, in seconds, for initial interaction.
Notes: If set to 0, the reader-defined default timeout is used. 0 by default.
(Read and Write property)

14.16.10 interactionTimeout as Double

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The timeout, in seconds, after the first key stroke.
Notes: If set to 0, the reader-defined default timeout is used. 0 by default.
(Read and Write property)

14.16.11 Events

14.16.12 characterEnteredInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that a valid character has been entered.

14.16.13 correctionKeyPressedInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that a correction key has been pressed.

14.16.14 invalidCharacterEnteredInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that an invalid character has been entered.

14.16.15 newPINConfirmationRequestedInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that the new PIN needs to be re-entered for confirmation.

14.16.16 newPINRequestedInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that the new PIN needs to be entered.

14.16.17 oldPINRequestedInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that the old PIN needs to be entered.

14.16.18 runCompleted(success as boolean, error as NSErrorMBS)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Called when run method is done.
Notes: success: Whether the user interaction was successful.
error: Contains information about the the error that occurred during the user interaction.
The NSError object is created in the TKErrorDomain domain with a code in the TKErrorCode enumeration.

14.16.19 validationKeyPressedInUserInteraction

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that the validation key has been pressed, indicating the end of PIN entry.

14.17 class TKTLVRecordMBS

14.17.1 class TKTLVRecordMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The base class encapsulating a Tag-Length-Value record.
Notes: The CryptoTokenKit framework provides the following concrete subclasses for various TLV record encodings:

- TKBERTLVRecordMBS for BER-TLV encoding rules
- TKSimpleTLVRecordMBS for Simple-TLV encoding according to ISO 7816-4
- TKCompactTLVRecordMBS for Compact-TLV encoding according to ISO 7816-4

Requires macOS 10.12 or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

14.17.2 Methods

14.17.3 available as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on macOS 10.12 or newer.

14.17.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

14.17.5 recordFromData(data as MemoryBlock) as TKTLVRecordMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates and returns a TLV record from by parsing the specified data.
Notes: data: A data object containing the serialized representation of a TLV record.

Returns a TLV record, or nil if data does not specify a valid record.

14.17.6 sequenceOfRecordsFromData(data as MemoryBlock) as TKTLVRecordMBS()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates and returns an array of TLV records from the specified data.
Notes: data: A data object containing the serialized representation of zero or more TLV records.

Returns a sequence of TLV records, or nil if data does not specify a sequence of valid records.

14.17.7 Properties

14.17.8 Data as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The record data, including the tag, length, and value fields.
Notes: (Read only property)

14.17.9 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.17.10 Tag as UInt64

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The tag field of the record.
Notes: (Read only property)

14.17.11 Value as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The value field of the record.
Notes: (Read only property)

14.18 class TKTokenAuthOperationMBS

14.18.1 class TKTokenAuthOperationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An authentication operation for a cryptographic token.
Notes: The CryptoTokenKit framework provides the following concrete subclasses: TKTokenPasswordAuthOperationMBS, for password-based authentication, and TKTokenSmartCardPINAuthOperationMBS for Smart Card PIN-based authentication.

14.18.2 Methods

14.18.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.18.4 finishWithError(byref error as NSErrorMBS) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Finishes the authentication operation.
Notes: error: On return, if an error occurred, contains an object with details of the error.
The NSError object is created in the TKErrorDomain domain with a code in the TKErrorCode enumeration.

Return true if the authentication operation finished successfully; otherwise false.

14.18.5 Properties

14.18.6 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.19 class TKTokenDriverMBS

14.19.1 class TKTokenDriverMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The abstract base class for building token drivers.
Notes: When working with Smart Card tokens, use or inherit from the TKSmartCardTokenDriverMBS subclass instead.

14.19.2 Methods

14.19.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.19.4 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.19.5 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets delegate, so events fire.
Notes: Called by constructor for you.

14.19.6 Properties

14.19.7 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.19.8 Events

14.19.9 createTokenForSmartCard(smartCard as TKSmartCardMBS, AID as MemoryBlock, byref error as NSErrorMBS) as TKSmartCardTokenMBS

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that a new Smart Card is detected.
Notes: self: The Smart Card token driver.
smartCard: The detected Smart Card.
AID: The ISO 7816-4 application identifier that is selected on the Smart Card. If the com.apple.ctk.aid attributes is not present in the Smart Card token extension property list, no application is selected.
error: If an error occured, this parameter should be populated with an NSError object containing details about the error.

Return the token created for the Smart Card, or nil if an error occurs or the delegate decides not to provide a token.

This event is only for TKSmartCardTokenDriverMBS class.

14.19.10 terminateToken(token as TKTokenMBS)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to terminate the specified token.

14.20 class TKTokenKeyAlgorithmMBS

14.20.1 class TKTokenKeyAlgorithmMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Cryptographic algorithms used by token keys.
Notes: Typically, the supported algorithm for a token key can be represented by a value of the SecKeyAlgorithm enumeration. However, tokens such as Smart Cards require that input data for operations take the format of a more specific algorithm. For example, a token may accept raw data to generate a cryptographic signature, but require that raw data to be formatted according to PKCS1 padding rules. To express such a requirement, a TKTokenKeyAlgorithm object defines a target algorithm and a set of other algorithms that were used. In the previous example, the target algorithm is kSecKeyAlgorithmRSASignatureRaw and the kSecKeyAlgorithmRSASignatureDigestPKCS1v15SHA1 algorithm is also reported as being used.

Requires macOS 10.12+ or newer.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

14.20.2 Methods

14.20.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

14.20.4 isAlgorithm(algorithm as string) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns whether the specified algorithm is the target operation algorithm.
Notes: algorithm: The algorithm to be checked.

Returns true if algorithm is the target operation algorithm; otherwise, false.

14.20.5 supportsAlgorithm(algorithm as string) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the specified algorithm is the target operation algorithm, or one of the other algorithms used.

Notes: algorithm: The algorithm to be checked.

Returns true if algorithm is the target operation algorithm or one of the other algorithms used; otherwise, false.

14.20.6 Properties

14.20.7 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.21 class TKTokenKeychainCertificateMBS

14.21.1 class TKTokenKeychainCertificateMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A token,Äôs certificate as stored in the keychain.
Notes: Subclass of the TKTokenKeychainItemMBS class.

14.21.2 Methods

14.21.3 Constructor(CertificateData as MemoryBlock, TKTokenObjectID as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token keychain certificate with data from the specified certificate reference and a given object ID.
Notes: CertificateData: The certificate as data block.
objectID: The object ID.

Returns a new token keychain certificate.
See also:

- 14.21.4 Constructor(TKTokenObjectID as Variant) 1221

14.21.4 Constructor(TKTokenObjectID as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.21.3 Constructor(CertificateData as MemoryBlock, TKTokenObjectID as Variant) 1221

14.21.5 Properties

14.21.6 data as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns a DER-encoded representation of an X.509 certificate.
Notes: (Read only property)

14.22 class TKTokenKeychainContentsMBS

14.22.1 class TKTokenKeychainContentsMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of the state of the keychain for a particular token.
Notes: Available on macOS 10.12+.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

14.22.2 Methods

14.22.3 certificateForObjectID(TKTokenObjectID as Variant, byref error as NSErrorMBS) as TKTokenKeychainCertificateMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the key for a specified object identifier.
Notes: objectID: The object identifier for the keychain item.
error: On return, if no such certificate exists, contains information about the error.

Returns the certificate, or nil if no certificate exists.

14.22.4 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

14.22.5 fillWithItems(items() as TKTokenKeychainItemMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Fills the keychain with the specified items.
Notes: items: The items to be added to the keychain.

All existing items for the token are first removed from the keychain before filling the keychain with items.

14.22.6 items as TKTokenKeychainItemMBS()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns all items for token in the keychain.

14.22.7 keyForObjectID(TKTokenObjectID as Variant, byref error as NSErrorMBS) as TKTokenKeychainKeyMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the key for a specified object identifier.
Notes: objectID: The object identifier for the keychain item.
error: On return, if no such key exists, contains information about the error.

Returns the key, or nil if no key exists.

14.22.8 Properties

14.22.9 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.23 class TKTokenKeychainItemMBS

14.23.1 class TKTokenKeychainItemMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An abstract base class for managing a token,Äôs contents as keychain items.
Notes: Don't use this base class directly. Instead, use one of its subclasses, such as TKTokenKeychainCertificateMBS for managing certificates or TKTokenKeychainKeyMBS for managing cryptographic keys.
Available on macOS 10.12+.

14.23.2 Methods

14.23.3 Constructor(TKTokenObjectID as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token keychain item with the specified object ID.

14.23.4 Properties

14.23.5 constraints as Dictionary

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Access constraints for the keychain item, keyed by TKTokenOperation values wrapped in integers.
Notes: (Read and Write property)

14.23.6 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.23.7 Label as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The user-visible label for the keychain item.
Notes: This property is equivalent to the kSecAttrLabel attribute type.
(Read and Write property)

14.23.8 objectID as Variant

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the object ID used for keychain item identification.
Notes: (Read only property)

14.24 class TKTokenKeychainKeyMBS

14.24.1 class TKTokenKeychainKeyMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A token's key as stored in the keychain.
Notes: Available on macOS 10.12+.
Subclass of the TKTokenKeychainItemMBS class.
Blog Entries

- MBS Xojo Plugins, version 23.2pr5

14.24.2 Methods

14.24.3 Constructor(CertificateData as MemoryBlock, TKTokenObjectID as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token keychain key with data from the specified certificate and a given object ID.
Notes: certificateData: The certificate data.
objectID: The object ID.

Returns a new token keychain certificate.
See also:

- 14.24.4 Constructor(TKTokenObjectID as Variant) 1227

14.24.4 Constructor(TKTokenObjectID as Variant)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.24.3 Constructor(CertificateData as MemoryBlock, TKTokenObjectID as Variant) 1227

14.24.5 Properties

14.24.6 applicationTag as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private tag data.
Notes: This property is equivalent to the kSecAttrApplicationTag type attribute.
(Read and Write property)

14.24.7 canDecrypt as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the key can be used to decrypt data.
Notes: This property is equivalent to the kSecAttrCanDecrypt type attribute.
(Read and Write property)

14.24.8 canPerformKeyExchange as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the key can be used to perform Diffie-Hellman style cryptographic key exchange.
Notes: (Read and Write property)

14.24.9 canSign as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the key can be used to sign data.
Notes: (Read and Write property)

14.24.10 keySizeInBits as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The key size in bits.
Notes: (Read and Write property)

14.24.11 keyType as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The type of the key.
Notes: Currently, only kSecAttrKeyTypeRSA and kSecAttrKeyTypeECSECPrimeRandom are supported values.

(Read and Write property)

14.24.12 publicKeyData as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The public key data.
Notes: (Read and Write property)

14.24.13 publicKeyHash as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The SHA1 hash of the raw public key.
Notes: This property is equivalent to the kSecAttrApplicationLabel type attribute.
(Read and Write property)

14.24.14 SuitableForLogin as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Whether the key can be used for system login.
Notes: (Read and Write property)

14.25 class TKTokenKeyExchangeParametersMBS

14.25.1 class TKTokenKeyExchangeParametersMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Parameters used to perform specific key exchange operations.
Notes: Available on macOS 10.12+.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

14.25.2 Methods

14.25.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

14.25.4 Properties

14.25.5 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.25.6 requestedSize as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the requested output size, in bytes, of key exchange result.
Notes: This property should be ignored if the output size is not configurable for the specified key exchange algorithm.
(Read only property)

14.25.7 sharedInfo as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns shared information typically used during the key derivation (KDF) step of a key exchange algorithm.
Notes: This property should be ignored if shared information isn,Äôt used by the specified key exchange algorithm.
(Read only property)

14.26 class TKTokenMBS

14.26.1 class TKTokenMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A representation of a hardware-based cryptographic token.
Notes: When working with Smart Card tokens, use or inherit from the TKSmartCardToken subclass instead.
Available in macOS 10.12+.

14.26.2 Methods

14.26.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.26.4 Constructor(tokenDriver as TKTokenDriverMBS, instanceID as String) 1232

14.26.4 Constructor(tokenDriver as TKTokenDriverMBS, instanceID as String)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token using the specified driver.
Notes: tokenDriver: The driver of the token.
instanceID: A unique, persistent identifier for this token. This value is typically generated from the serial number of the target hardware.

Returns a new token object.
See also:

- 14.26.3 Constructor 1232

14.26.5 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.26.6 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the delegate, so you receive events.
Notes: Called by Constructor for you.

14.26.7 Properties

14.26.8 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.26.9 keychainContents as TKTokenKeychainContentsMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The contents of the keychain for this token.
Notes: (Read only property)

14.26.10 tokenDriver as TKTokenDriverMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The token driver.
Notes: (Read only property)

14.26.11 Events

14.26.12 createSession(byref error as NSErrorMBS) as TKTokenSessionMBS

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to create a session for the specified token.
Notes: token: The token.
error: If an error occurred, this method should return nil and populate this parameter with an object containing details of the error. The error object should have a domain equal to TKErrorDomain and a code

equal to a value defined by the TKErrorCode enumeration.

Return a new token session, or nil if an error occurred.

All operations for a token are performed within a session representing an authentication context. This delegate method is called whenever new authentication context is needed. For example, a client may want to perform a token operation using a keychain object that has an associated LAContext.

14.26.13 terminateSession(session as TKTokenSessionMBS)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to terminate the specified token session.

14.27 class TKTokenPasswordAuthOperationMBS

14.27.1 class TKTokenPasswordAuthOperationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An authentication operation for a cryptographic token.
Notes: The CryptoTokenKit framework provides the following concrete subclasses: TKTokenPasswordAuthOperationMBS, for password-based authentication, and TKTokenSmartCardPINAuthOperationMBS for Smart Card PIN-based authentication.
Subclass of the TKTokenAuthOperationMBS class.

14.27.2 Methods

14.27.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.27.4 Properties

14.27.5 password as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The password to be filled in when the finishWithError method is called.
Notes: (Read and Write property)

14.28 class TKTokenSessionMBS

14.28.1 class TKTokenSessionMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A token session that manages the authentication state of a token.
Notes: A token session communicates with its delegate to perform operations with its token that are bound to the authentication state.

A session is always instantiated by a TKTokenMBS instance through the token's delegate when the framework detects access to the token from a new authentication session.

Never share the authentication status of a token, such as the PIN entered to unlock a smart card, with other token sessions.

14.28.2 Methods

14.28.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 14.28.4 Constructor(token as TKTokenMBS) 1236

14.28.4 Constructor(token as TKTokenMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token session with the specified token.
Notes: token: The token to which the initialized session is bound.

Returns a new token session created with the specified token.
See also:

- 14.28.3 Constructor 1236

14.28.5 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.28.6 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the delegat, so events fire.
Notes: Called for you by constructor.

14.28.7 Properties

14.28.8 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.28.9 token as TKTokenMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The token to which the session is bound.
Notes: (Read only property)

14.28.10 Events

14.28.11 beginAuthForOperation(operation as Integer, constraint as Variant, byref error as NSErrorMBS) as TKTokenAuthOperationMBS

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate that authentication has begun for the specified operation and constraint.
Notes: self: The token session.
operation: The kind of operation.
constraint: The constraint to be satisfied.
error: If an error occurred, this method should return nil and populate this parameter with an object containing details of the error. The error object should have a domain equal to TKErrorDomain and a code equal to a value defined by the TKErrorCode enumeration.

Return the resulting context of the operation, or nil if an error occurred.

If you return an instance of a subclass of TKTokenAuthOperation that is provided by the CryptoTokenKit framework, the system will first fill in the context-specific properties, such as the password, before calling the finishWithError: method on the context.

14.28.12 decryptData(ciphertext as MemoryBlock, keyObjectID as Variant, algorithm as TKTokenKeyAlgorithmMBS, byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to decrypt a data object using the specified key and algorithm.
Notes: self: The token session.
ciphertext: The data to decrypt.
keyObjectID: The identifier of the public key object.
algorithm: The algorithm to be used for decryption.
error: If an error occurred, this method should return nil and populate this parameter with an object containing details of the error. The error object should have a domain equal to TKErrorDomain and a code equal to a value defined by the TKErrorCode enumeration.

Returns the decrypted data, or nil if an error occurred.

14.28.13 performKeyExchangeWithPublicKey(otherPartyPublicKeyData as MemoryBlock, keyObjectID as Variant, algorithm as TKTokenKeyAlgorithmMBS, parameters as TKTokenKeyExchangeParametersMBS, byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to perform a key exchange using the specified key and algorithm.
Notes: self: The token session.
otherPartyPublicKeyData: The public key of the other party.
keyObjectID: The identifier of the private key object.
algorithm: The algorithm to be used for key exchange.
parameters: Additional parameters used by algorithm to perform the key exchange.
error: If an error occurred, this method should return nil and populate this parameter with an object containing details of the error. The error object should have a domain equal to TKErrorDomain and a code equal to a value defined by the TKErrorCode enumeration.

Returns the result of the key exchange, or nil if an error occurred.

14.28.14 signData(dataToSign as MemoryBlock, keyObjectID as Variant, algorithm as TKTokenKeyAlgorithmMBS, byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Tells the delegate to sign a data object using the specified key and algorithm.
Notes: self: The token session.
data: The data to sign.
keyObjectID: The identifier of the private key object.
algorithm: The algorithm to be used for signing.
error: If an error occurred, this method should return nil and populate this parameter with an object containing details of the error. The error object should have a domain equal to TKErrorDomain and a code equal to a value defined by the TKErrorCode enumeration.

Return the signed data, or nil if an error occurred.

14.28.15 supportsOperation(operation as Integer, keyObjectID as Variant, algorithm as TKTokenKeyAlgorithmMBS) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Asks the delegate whether the token session supports a given operation using the specified key and algorithm.
Notes: self: The token session.
operation: The operation to perform. For possible values, see kTokenOperation* constants.
keyObjectID: The identifier of the private key object.
algorithm: The algorithm to be used by the operation.

Returns true if the operation is supported; otherwise, false.

14.28.16 Constants

Token Operations

Constant	Value	Description
kTokenOperationDecryptData	3	Decrypt data using a private key.
kTokenOperationNone	0	No operation.
kTokenOperationPerformKeyExchange	4	Perform a Diffie-Hellman style cryptographic key exchange using a priva
kTokenOperationReadData	1	Read raw data of a certificate.
kTokenOperationSignData	2	Create a cryptographic signature using a private key.

14.29 class TKTokenSmartCardPINAuthOperationMBS

14.29.1 class TKTokenSmartCardPINAuthOperationMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A Smart Card PIN authentication operation.
Notes: Subclass of the TKTokenAuthOperationMBS class.

14.29.2 Methods

14.29.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The constructor.

14.29.4 Properties

14.29.5 APDUTemplate as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The template into which the PIN is filled in. nil by default.
Notes: If nil, the system will not attempt to authenticate by sending the formatted APDU to the Smart Card. Instead, the token itself is expected to perform the authentication. You are encouraged to provide an APDU template, if possible, as it allows the use of a hardware interface for secure PIN entry, provided one exists.
(Read and Write property)

14.29.6 PIN as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The PIN value resulting from performing the operation.
Notes: This property is set to the result of the operation after finishWithError: is called.

If the APDUTemplate property has a set value, this property is not set, as the PIN is automatically sent to the Smart Card using the specified template.
(Read and Write property)

14.29.7 PINByteOffset as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The offset, in bytes, within the APDU template to mark the location for filling in the PIN.
Notes: (Read and Write property)

14.29.8 PINFormat as TKSmartCardPINFormatMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The PIN format.
Notes: By default, this property is set to a TKSmartCardPINFormatMBS object initialized without any further configuration.
(Read and Write property)

14.29.9 smartCard as TKSmartCardMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A Smart Card to which the formatted APDU is sent in order to authenticate.
Notes: This property is only used if the APDUTemplate property has a set value.
(Read and Write property)

14.30 class TKTokenWatcherMBS

14.30.1 class TKTokenWatcherMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: An object that tracks the tokens available in the system.
Notes: Create a token watcher and register an insertion handler to be notified when tokens are added to the system. You can also add removal handlers for specific tokens to be notified when those tokens are removed from the system.

14.30.2 Methods

14.30.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Initializes a token watcher.

14.30.4 Destructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The destructor.

14.30.5 SetDelegate

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the observer to call tokenIDsChanged event.
Notes: Called for you by constructor.

14.30.6 tokenIDs as String()

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The token IDs currently available in the system.
Notes: Each string in tokenIDs corresponds to the name of the token instance.

14.30.7 Properties

14.30.8 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

14.30.9 Events

14.30.10 Inserted(tokenID as string)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: The event called when the given token is inserted.

14.30.11 Removed(tokenID as string)

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: The event called when the given token is removed.

14.30.12 tokenIDsChanged

Plugin Version: 18.5, Platform: macOS, Targets: .

Function: Event called when tokenID list changed.

Chapter 15

Social

15.1 class ACAccountCredentialMBS

15.1.1 class ACAccountCredentialMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: An ACAccountCredential object encapsulates the information needed to authenticate a user.
Notes: To create an account credential that uses the OAuth open authentication standard, use the Constructor.

15.1.2 Methods

15.1.3 Constructor(token as string, refreshToken as string, expiryDate as date)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes an account credential using OAuth 2.
Notes: token: The client application,Äôs token.
refreshToken: The client application,Äôs refresh token.
expiryDate: The date the token expires.

Accounts can optionally use the OAuth open authentication standard to authenticate your client application. Instead of the user giving their username and password to log in, the server authenticates the user, and your client application receives a token that grants it access to specific resources for a defined duration. The authentication mechanism uses a key and secret scheme similar to the public and private keys used by ssh. A token is a unique, random string of letters and numbers that is paired with a secret to protect the token from being abused. You initialize account credentials using this token and secret token.

To learn more about OAuth, go to Hueniverse OAuth.
http://hueniverse.com/oauth/
See also:

- 15.1.4 Constructor(token as string, refreshToken as string, expiryDate as dateTime) 1246
- 15.1.5 Constructor(token as string, tokenSecret as string) 1246

15.1.4 Constructor(token as string, refreshToken as string, expiryDate as dateTime)

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Initializes an account credential using OAuth 2.
See also:

- 15.1.3 Constructor(token as string, refreshToken as string, expiryDate as date) 1245
- 15.1.5 Constructor(token as string, tokenSecret as string) 1246

15.1.5 Constructor(token as string, tokenSecret as string)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Initializes an account credential using OAuth.
Notes: Accounts can optionally use the OAuth open authentication standard to authenticate your client application. Instead of the user giving their username and password to log in, the server authenticates the user, and your client application receives a token that grants it access to specific resources for a defined duration. The authentication mechanism uses a key and secret scheme similar to the public and private keys used by ssh. A token is a unique, random string of letters and numbers that is paired with a secret to protect the token from being abused. You initialize account credentials using this token and secret token.

To learn more about OAuth, go to Hueniverse OAuth.
http://hueniverse.com/oauth/
See also:

- 15.1.3 Constructor(token as string, refreshToken as string, expiryDate as date) 1245
- 15.1.4 Constructor(token as string, refreshToken as string, expiryDate as dateTime) 1246

15.1.6 Properties

15.1.7 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

15.1.8 oauthToken as String

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The token used for the credential.
Notes: (Read and Write property)

15.2 class ACAccountMBS

15.2.1 class ACAccountMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The account class.
Notes: An ACAccount object encapsulates information about a user account stored in the Accounts database. You can create and retrieve accounts using an ACAccountStore object. The ACAccountStore object provides an interface to the persistent Accounts database. For each user, all account objects belong to a single ACAccountStore object.

Available in OS X v10.8 and later in 64-bit applications.

15.2.2 Methods

15.2.3 Constructor(type as ACAccountTypeMBS)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Initializes a new account of the specified type.
Notes: Available in OS X v10.8 and later.

15.2.4 Destructor

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The destructor.

15.2.5 Properties

15.2.6 accountDescription as String

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A human-readable description of the account.
Notes: This property is available if the user grants the application access to this account; otherwise it is "". (Read and Write property)

15.2.7 accountType as ACAccountTypeMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The type of service account.
Notes: This property is required. You specify the account type using the Constructor. You can use the accountsWithAccountType method to retrieve all accounts of a particular type.
(Read and Write property)

15.2.8 credential as ACAccountCredentialMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The credential used to authenticate the user of this account.
Notes: This property is required and must be set before the account is saved. For privacy reasons, this property is inaccessible after the account is saved.
(Read and Write property)

15.2.9 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Internal object reference.
Notes: (Read and Write property)

15.2.10 identifier as String

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A unique identifier for this account.
Notes: Use the accountWithIdentifier method to get an account with the specified identifier.
(Read only property)

15.2.11 Parent as ACAccountStoreMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The account store for this account.
Notes: (Read only property)

15.2.12 username as String

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The username for this account.
Notes: This property must be set before the account is saved. After the account is saved, this property is available if the user grants the application access to this account; otherwise it is "".
(Read and Write property)

15.3 class ACAccountStoreMBS

15.3.1 class ACAccountStoreMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The ACAccountStore class provides an interface for accessing, manipulating, and storing accounts.
Notes: To create and retrieve accounts from the Accounts database, you must create an ACAccountStore object. Each ACAccount object belongs to a single ACAccountStore object.

Available on Mac OS X 10.8 and later in 64-bit applications.

15.3.2 Methods

15.3.3 ACAccountStoreDidChangeNotification as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: One of the notification names.
Notes: Posted when the accounts managed by this account store changed in the database. There is no userInfo dictionary associated with this notification.

This notification is sent if an account is saved or removed locally or externally. If you receive this notification, you should refetch all account objects.

The plugin automatically registeres this for the Changed event.

15.3.4 accounts as ACAccountMBS()

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The accounts managed by this account store.

15.3.5 accountsWithAccountType(type as ACAccountTypeMBS) as ACAccountMBS()

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns all accounts of the specified type.

15.3.6 accountTypeWithAccountTypeIdentifier(identifier as string) as ACAccountTypeMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns an account type that matches the specified identifier.
Notes: typeIdentifier: The account type identifier.

15.3.7 accountWithIdentifier(identifier as string) as ACAccountMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns the account with the specified identifier.

15.3.8 ACErrorDomain as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The error domain for account framework.

15.3.9 available as boolean

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Whether class is available.
Notes: Returns true on Mac OS X 10.8 in a 64-bit application.

15.3.10 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The constructor.

15.3.11 Destructor

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The destructor.

15.3.12 removeAccount(account as ACAccountMBS, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Removes an account from the account store.
Notes: account: The account to remove.

This call will fail if you don't have sufficient rights to remove the account.

15.3.13 renewCredentialsForAccount(account as ACAccountMBS, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Renews account credentials when the credentials are no longer valid.
Notes: account: The account to renew credentials.

For Twitter and Sina Weibo accounts, this method will prompt the user to go to Settings to re-enter their password.
For Facebook accounts, if the access token has become invalid due to a regular expiration, this method will obtain a new one.
If the user has deauthorized your app, this renewal request will return ACAccountCredentialRenewResultRejected.

15.3.14 requestAccessToAccountsWithType(accountType as ACAccountTypeMBS, dic as dictionary, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Obtains permission to access protected user properties.
Notes: accountType: The account type.
options: The account options.

Certain account types (such as Facebook) require an options dictionary. This method will throw an NSInvalidArgumentException if the options dictionary is not provided for such account types. Conversely, if the account type does not require an options dictionary, the options parameter must be nil.

15.3.15 saveAccount(account as ACAccountMBS, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Saves an account to the Accounts database.
Notes: account: The account to save.

If the account type supports authentication and the account is not authenticated, the account server uses the account's credentials to authenticate it. If the authentication is successful, the account is saved; otherwise it is not saved.

15.3.16 Properties

15.3.17 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

15.3.18 Events

15.3.19 Changed

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: Event called when database changed.
Notes: Posted when the accounts managed by this account store changed in the database. There is no userInfo dictionary associated with this notification.

This notification is sent if an account is saved or removed locally or externally. If you receive this notification, you should refetch all account objects.

15.3.20 removeAccountCompleted(account as ACAccountMBS, success as boolean, error as NSErrorMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The event called when account removing completed.

15.3.21 renewCredentialsForAccountCompleted(account as ACAccountMBS, renewResult as Integer, error as NSErrorMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: Event called when renew credentials completed.

15.3.22 requestAccessCompleted(granted as boolean, error as NSErrorMBS, accountType as ACAccountTypeMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The event called when requestAccessToAccountsWithType completed.

15.3.23 saveAccountCompleted(success as boolean, error as NSErrorMBS, account as ACAccountMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The account save completed.

15.3.24 Constants

Renew Status Codes

Constant	Value	Description
ACAccountCredentialRenewResultFailed	2	A non-user-initiated cancel of the prompt. Try again.
ACAccountCredentialRenewResultRejected	1	Renewal failed because the user revoked your access to their accou
ACAccountCredentialRenewResultRenewed	0	The account,Äôs credentials have been renewed and are now asso the account.

Errors

Constant	Value	Description
ACErrorAccessDeniedByProtectionPolicy	10	Due to the current protection policy in effect, we couldn't fetch a cr
ACErrorAccessInfoInvalid	8	The client's access info dictionary has incorrect or missing values.
ACErrorAccountAlreadyExists	5	Account wasn't added because it already exists.
ACErrorAccountAuthenticationFailed	3	Account wasn't saved because authentication of the supplied credent
ACErrorAccountMissingRequiredProperty	2	Account wasn't saved because it is missing a required property.
ACErrorAccountNotFound	6	Account wasn't deleted because it could not be found.
ACErrorAccountTypeInvalid	4	Account wasn't saved because the account type is invalid.
ACErrorClientPermissionDenied	9	Your client does not have access to the requested data.
ACErrorCredentialNotFound	11	Yo, I tried to find your credential, but it must have run off!
ACErrorFetchCredentialFailed	12	Something bad happened on the way to the keychain
ACErrorInvalidClientBundleID	16	The client making the request does not have a valid bundle ID.
ACErrorPermissionDenied	7	The operation didn't complete because the user denied permission.
ACErrorRemoveCredentialFailed	14	Unable to remove credential
ACErrorStoreCredentialFailed	13	Unable to store credential
ACErrorUnknown	1	Unknown error
ACErrorUpdatingNonexistentAccount	15	Account save failed because the account being updated has been ren

15.4 class ACAccountTypeMBS

15.4.1 class ACAccountTypeMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: An ACAccountType object encapsulates information about all accounts of a particular type.
Notes: You do not create account type objects directly. To obtain an account type, use the Constructor or the accountType property of an account object. Account Type Identifiers describes the identifiers for currently supported account types. You can also use the accountsWithAccountType method to obtain all accounts of a particular type.

Available in OS X v10.8 and later in 64-bit applications.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 18.5pr3

15.4.2 Methods

15.4.3 ACAccountTypeIdentifierFacebook as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Identifier for the Facebook account type.

15.4.4 ACAccountTypeIdentifierLinkedIn as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Identifier for the LinkedIn account type.
Notes: Available in OS X v10.9 and later.

15.4.5 ACAccountTypeIdentifierSinaWeibo as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Identifier for the Sina Weibo account type.

15.4.6 ACAccountTypeIdentifierTencentWeibo as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Identifier for the Tencent Weibo account type.
Notes: Available in OS X v10.9 and later.

15.4.7 ACAccountTypeIdentifierTwitter as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Identifier for the Twitter account type.

15.4.8 accessGranted as boolean

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the user granted the application access to accounts of this type.
Notes: True if the application has access to accounts of this type; otherwise false.

15.4.9 accountTypeDescription as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A human-readable description of the account type.

15.4.10 ACFacebookAppIdKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Your Facebook App ID, as it appears on the Facebook website.

15.4.11 ACFacebookAudienceEveryone as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Posts from your app are visible to everyone.

15.4.12 ACFacebookAudienceFriends as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Posts are visible only to friends.

15.4.13 ACFacebookAudienceKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Only required when posting permissions are requested.

15.4.14 ACFacebookAudienceOnlyMe as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Posts are visible to the user only.

15.4.15 ACFacebookPermissionsKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: An array of of the permissions you're requesting.

15.4.16 ACLinkedInAppIdKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Your LinkedIn App ID (or API Key), as it appears on the LinkedIn website.

15.4.17 ACLinkedInPermissionsKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: An array of of the LinkedIn permissions you're requesting.

15.4.18 ACTencentWeiboAppIdKey as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Tencent App ID

15.4.19 Constructor

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The private constructor.

15.4.20 identifier as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The unique identifier for the account type.

15.4.21 Properties

15.4.22 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

15.5 class SLRequestMBS

15.5.1 class SLRequestMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The SLRequest object encapsulates the properties of an HTTP request, providing a convenient template for you to make requests.
Notes: You send a request to a social networking service to perform some operation on behalf of the user or to retrieve user information.

HTTP requests have these common components: an HTTP request method (GET, POST, PUT, or DELETE), a URL identifying the operation to perform, a set of query parameters, and an optional multipart POST body containing additional data. The values for these properties depend on the request you are sending and the target service provider. Refer to each supported social networking site,Äôs documentation for possible values. Links to documentation are provided in Table 1.

Use the requestForServiceType:requestMethod:URL:parameters: method to initialize a newly created SLRequest object passing the required property values. Use the addMultipartData:withName:type: to optionally specify a multipart POST body. After you create your request, use the performRequestWithHandler: method to send the request, specifying the handler to call when the request is done.

If you already have a sending mechanism, you can use the preparedURLRequest method to create the request that you send using an NSURLConnection object. If the request requires user authorization, set the account property to an ACAccount object.

Available in OS X v10.8 and later. 64bit only.

15.5.2 Methods

15.5.3 addMultipartData(data as memoryblock, name as string, type as string, filename as string)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Specifies a named multipart POST body for this request.
Notes: data: The data for the multipart POST body, such as an image or text.
name: The name of the multipart POST body. This is the name that a specific social service expects.
type: The type of the multipart POST body. This is the MIME content type of the multipart data.
filename: The filename of the attachment that you want to POST. Many social services require a filename in order to accept certain POST requests, such as uploading an image or video. If your multipart data does not require a filename, pass in nil.

Possible parameter values are dependent on the target service. This information, as well as guidance on when to use a multipart POST body, is documented by the service provider.

15.5.4 Available as boolean

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on OS X 10.8 and newer in a 64-bit app.

15.5.5 Constructor(serviceType as string, requestMethod as Integer, URL as string, parameters as dictionary)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Initializes a newly created request object with the specified properties.
Notes: serviceType: The social networking service type. e.g. SLServiceTypeTwitter
requestMethod: The method to use for this HTTP request. e.g. SLRequestMethodPOST
url: The destination URL for this HTTP request. The values and formatting for the URL are dependent on the target service and are documented by the service provider.
parameters: The parameters for this HTTP request. The values and formatting are dependent on the target service and are documented by the service provider.

The newly initialized request object.
Use this method to initialize an SLRequest. The value and formatting of each parameter is dependent on the target service.

15.5.6 performRequest(tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Performs an asynchronous request and calls the RequestHandler event when done.

15.5.7 preparedURLRequest as NSURLRequestMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns an authorized URL request that can be sent using an NSURLConnectionMBS object.
Notes: An OAuth-compatible NSURLRequest object that allows an app to act on behalf of the user while keeping the user,Äôs password private. The NSURLRequest is signed as OAuth1 by default, or OAuth2 by

adding the appropriate token based on the user,Äôs account.

Use this method to modify your request before sending. By setting the account correctly, this method will automatically add any necessary tokens.

15.5.8 SLServiceTypeFacebook as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A string constant that identifies the social networking site, Facebook.

15.5.9 SLServiceTypeLinkedIn as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A string constant that identifies the social networking site, LinkedIn.

15.5.10 SLServiceTypeSinaWeibo as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A string constant that identifies the social networking site, Sina Weibo.

15.5.11 SLServiceTypeTencentWeibo as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A string constant that identifies the social networking site, Tencent Weibo.

15.5.12 SLServiceTypeTwitter as string

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: A string constant that identifies the social networking site, Twitter.

15.5.13 Properties

15.5.14 account as ACAccountMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Account information used to authenticate the request.
Notes: The account is used to sign a request with OAuth1 services or to add an access token for OAuth2 services. By associating the account with the request, the necessary tokens are added automatically. The default value is nil.
(Read and Write property)

15.5.15 Handle as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

15.5.16 parameters as Dictionary

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The parameters for this request.
Notes: Use this property to look up the query parameters of the HTTP request that was set in Constructor. Possible values are dependent on the target service and are documented by the service provider.
(Read only property)

15.5.17 requestMethod as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The method to use for this request.
Notes: Use this property to look up the method of the HTTP request that was set in requestForServiceType.
(Read only property)

15.5.18 URL as String

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The destination URL for this request.
Notes: (Read only property)

15.5.19 Events

15.5.20 performRequestCompleted(responseData as memoryblock, urlResponse as NSURLResponseMBS, error as NSErrorMBS, tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: .

Function: The callback handler for a request..
Notes: responseData: The data returned by the request. The format of this data is dependent on the target service.
urlResponse: The URL response returned by the request that includes the HTTP response codes.
error: An error identifier.
Possible values are dependent on the target service and are documented by the service provider.

15.5.21 Constants

Constants

Constant	Value	Description
SLRequestMethodDELETE	2	One of the request types. Deletes the specified resource. Available in OS X v10.8 and later.
SLRequestMethodGET	0	One of the request types. Requests information from the specified resource. Use a GET request to fetch information from the specified server such as character limits or a user,Äôs timeline. Available in OS X v10.8 and later.
SLRequestMethodPOST	1	One of the request types. Submits data to be processed. Use a POST request to submit information to the specified server such as a status update or an image. Available in OS X v10.8 and later.
SLRequestMethodPUT	3	One of the request types. Uses a PUT request to submit the data. Available in OS X v10.9 and later.

Chapter 16

SpeechRecognition

Chapter 17

TouchBar

17.1 class LAContextMBS

17.1.1 class LAContextMBS

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: The class for local authentication via password or TouchID.
Blog Entries

- Xojo 2021r1
- MBS Xojo Plugins Version 21.0 News
- MBS Xojo Plugins, version 18.4pr3
- MonkeyBread Software Releases the MBS Xojo plug-ins in version 17.0
- MBS Xojo / Real Studio Plugins, version 17.0pr1

Videos

- Presentation from London conference about MBS Plugins.

Xojo Developer Magazine

- 15.5, page 35: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar
- 15.2, page 9: News

17.1.2 Methods

17.1.3 Available as Boolean

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Whether the class is available.
Notes: Returns true on OS X 10.10 and newer in 64-bit application.

17.1.4 canEvaluatePolicy(Policy as Integer, byref Error as NSErrorMBS) as Boolean

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Determines if a particular policy can be evaluated.
Example:

```
try
dim e as NSErrorMBS

if lc.canEvaluatePolicy(lc.PolicyDeviceOwnerAuthenticationWithBiometrics, e) then
List.AddRow "Can authenticate with biometrics."
else
List.AddRow "Can't authenticate with biometrics."
end if
if e <>nil then
List.AddRow "Error: "+e.LocalizedDescription
end if

catch ex as NSExceptionMBS

List.AddRow "Exception: "+ex.Message

end try
```

Notes: Policies can have certain requirements which, when not satisfied, would always cause the policy evaluation to fail. Examples can be a passcode set or a fingerprint enrolled with Touch ID. This method allows easy checking for such conditions.

Applications should consume the returned value immediately and avoid relying on it for an extensive period of time. At least, it is guaranteed to stay valid until the application enters background.

policy: Policy for which the preflight check should be run.

error: contains error information if policy evaluation is not possible.

Returns true if the policy can be evaluated, false otherwise.

17.1.5 Constructor

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: The constructor.

17.1.6 evaluatePolicy(Policy as Integer, localizedReason as String, Tag as Variant = nil)

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Evaluates the specified policy.
Example:

```
dim lc as LAContextMBS // your context

try

lc.evaluatePolicy(lc.PolicyDeviceOwnerAuthenticationWithBiometrics, "We need to test.")

catch ex as NSExceptionMBS

List.AddRow "Exception: "+ex.Message

end try
```

Notes: Policy evaluation may involve prompting user for various kinds of interaction or authentication. Actual behavior is dependent on evaluated policy, device type, and can be affected by installed configuration profiles.

Be sure to keep a strong reference to the context while the evaluation is in progress. Otherwise, an evaluation would be canceled when the context is being deallocated.

The method does not block. Instead the event is called asynchronously when evaluation finishes.

Implications of successful policy evaluation are policy specific. In general, this
operation is not idempotent. Policy evaluation may fail for various reasons, including

user cancel, system cancel and others, see error codes.

policy: Policy to be evaluated.
localizedReason: Application reason for authentication. This string must be provided in correct localization and should be short and clear. It will be eventually displayed in the authentication dialog subtitle. A name of the calling application will be already displayed in title, so it should not be duplicated here.

17.1.7 invalidate

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Invalidates the context.
Notes: The context is invalidated automatically when it is (auto)released. This method allows invalidating it manually while it is still in scope.

Invalidation terminates any existing policy evaluation and the respective call will fail with LAErrorAppCancel. After the context has been invalidated, it can not be used for policy evaluation and an attempt to do so will fail with LAErrorInvalidContext.

Invalidating a context that has been already invalidated has no effect.

17.1.8 isCredentialSet(CredentialType as Integer) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Reveals if credential was set with this context.
Notes: type: Type of credential we are asking for.

Return true on success, false otherwise.

17.1.9 setCredential(credential as MemoryBlock, Type as Integer) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Sets a credential to this context.
Notes: Some policies allow to bind application-provided credential with them.
This method allows credential to be passed to the right context.

credential: Credential to be used with subsequent calls. Setting this parameter to nil will remove any existing credential of the specified type.

Type: Type of the provided credential.

Return true if the credential was set successfully, false otherwise.

17.1.10 TouchIDAuthenticationMaximumAllowableReuseDuration as Double

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: The maximum value for LAContext touchIDAuthenticationAllowableReuseDuration property.
Notes: Returns -1 if value is not available.

17.1.11 Properties

17.1.12 biometryType as Integer

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Indicates the type of the biometry supported by the device.
Notes: This property is set when canEvaluatePolicy has been called for a biometric policy.
The default value is BiometryTypeNone.
(Read only property)

17.1.13 evaluatedPolicyDomainState as MemoryBlock

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Contains policy domain state.
Notes: This property is set only when evaluatePolicy is called and succesful Touch ID authentication was performed, or when canEvaluatePolicy succeeds for a biometric policy. It stays nil for all other cases. If finger database was modified (fingers were removed or added), evaluatedPolicyDomainState data will change. Nature of such database changes cannot be determined but comparing data of evaluatedPolicyDomainState after different evaluatePolicy will reveal the fact database was changed between calls.
(Read only property)

17.1.14 Handle as Integer

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: The internal object handle.
Notes: (Read and Write property)

17.1.15 interactionNotAllowed as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Allows running authentication in non-interactive mode.
Notes: If the context is used in a keychain query by the means of kSecUseAuthenticationContext, then setting this property to YES has the same effect as passing kSecUseNoAuthenticationUI in the query, i.e. the keychain call will eventually fail with errSecInteractionNotAllowed instead of displaying the authentication UI.

If this property is used with a LocalAuthentication evaluation, it will eventually fail with kErrorNotInteractive instead of displaying the authentication UI.
(Read and Write property)

17.1.16 localizedCancelTitle as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Cancel button title.
Notes: Allows cancel button title customization. A default title "Cancel" is used when this property is set to empty string.
Requires MacOS 10.13 or newer.
(Read and Write property)

17.1.17 localizedFallbackTitle as String

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Fallback button title.
Example:

```
dim lc as new MyLAContextMBS
lc.localizedFallbackTitle = "Just a test"
```

Notes: Allows fallback button title customization. A default title "Enter Password" is used when this property is left nil. If set to empty string, the button will be hidden.
(Read and Write property)

17.1.18 localizedReason as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Allows setting the default localized authentication reason on context.
Notes: A localized string from this property is displayed in the authentication UI if the caller didn't specify its own authentication reason (e.g. a keychain operation with kSecUseAuthenticationContext). This property is ignored if the authentication reason was provided by caller.
Requires MacOS 10.13 or newer.
(Read and Write property)

17.1.19 touchIDAuthenticationAllowableReuseDuration as Double

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: Time interval for accepting a successful Touch ID unlock from the past.
Notes: This property can be set with a time interval in seconds. If the device was successfully unlocked by Touch ID within this time interval, then Touch ID authentication on this context will succeed automatically and the reply block will be called without prompting user for Touch ID.

The default value is 0, meaning that no previous TouchID authentication can be reused.

The maximum supported interval is 5 minutes and setting the value beyond 5 minutes does not increase the accepted interval.
(Read and Write property)

17.1.20 Events

17.1.21 evaluatePolicyResult(Success as Boolean, error as NSErrorMBS, Policy as Integer, localizedReason as String, tag as Variant)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: The event called for the result of evaluatePolicy method.
Notes: success: Reply parameter that is true if the policy has been evaluated successfully or false if the evaluation failed.

error: Reply parameter that is nil if the policy has been evaluated successfully, or it contains error information about the evaluation failure.

Typical error codes returned by this call are:
ErrorUserFallback if user tapped the fallback button

ErrorUserCancel if user has tapped the Cancel button
ErrorSystemCancel if some system event interrupted the evaluation (e.g. Home button pressed).

17.1.22 Constants

Constants

Constant	Value	Description
kLAErrorDomain	”com.apple.LocalAuthentication”	The error domain for Local Authentication.

Biometry Types

Constant	Value	Description
BiometryTypeFaceID	2	The device supports Face ID.
BiometryTypeNone	0	The device does not support biometry.
BiometryTypeTouchID	1	The device supports Touch ID.

Credential Types

Constant	Value	Description
CredentialTypeApplicationPassword	0	The credential type for an application password.
CredentialTypeSmartCardPIN	1	Smart card PIN provided by application. If not set, LocalAuthentication will ask users for the smart card PIN w necessary. Applications can provide the PIN using setCredential method. such case, LocalAuthentication will not show the smart card PIN user interf When entered from the LocalAuthentication user interface, the PIN is sto as UTF-8 encoded string.

Error Codes

Constant	Value	Description
kErrorAppCancel	-9	Authentication was canceled by application (e.g. invalidate was called while authentication was in progress).
kErrorAuthenticationFailed	-1	Authentication was not successful, because user failed to provide valid credentials.
kErrorBiometryLockout	-8	Authentication was not successful, because there were too many failed Biometry attempts and Biometry is now locked. Passcode is required to unlock Biometry, e.g. evaluating PolicyDeviceOwnerAuthenticationWithBiometrics will ask for passcode as a prerequisite.
kErrorBiometryNotAvailable	-6	Authentication could not start, because Biometry is not available on the device.
kErrorBiometryNotEnrolled	-7	Authentication could not start, because Biometry has no enrolled fingers.
kErrorInvalidContext	-10	LAContext passed to this call has been previously invalidated.
kErrorPasscodeNotSet	-5	Authentication could not start, because passcode is not set on the device.
kErrorSystemCancel	-4	Authentication was canceled by system (e.g. another application went to foreground).
kErrorTouchIDLockout	-8	Authentication was not successful, because there were too many failed Touch ID attempts and Touch ID is now locked. Passcode is required to unlock Touch ID, e.g. evaluating PolicyDeviceOwnerAuthenticationWithBiometrics will ask for passcode as a prerequisite.
kErrorTouchIDNotAvailable	-6	Authentication could not start, because Touch ID is not available on the device.
kErrorTouchIDNotEnrolled	-7	Authentication could not start, because Touch ID has no enrolled fingers.
kErrorUserCancel	-2	Authentication was canceled by user (e.g. tapped Cancel button).
kErrorUserFallback	-3	Authentication was canceled, because the user tapped the fallback button (Enter Password).

Policies

Constant	Value	Description
PolicyDeviceOwnerAuthentication	2	Device owner is going to be authenticated by biome Touch ID or user password authentication is requ available, not enrolled or locked out, then the user i away. Touch ID authentication dialog behaves similarly cyDeviceOwnerAuthenticationWithBiometrics. How button does not end the authentication. Instead, it s mechanism to user password.
PolicyDeviceOwnerAuthenticationWithBiometrics	1	The policy for owner authentication with biometrics
PolicyDeviceOwnerAuthenticationWithBiometricsOrWatch	4	Device owner is going to be authenticated by biome Watch or biometric authentication is required. If no can be found, it behaves as PolicyDeviceOwnerAuth Similarly, if biometry is unavailable it behaves as P ticationWithWatch. Watch authentication dialog looks and behaves simi When both mechanisms are available, user is asked authentication will run in parallel.
PolicyDeviceOwnerAuthenticationWithWatch	3	Watch authentication is required. If no nearby pa found, kErrorWatchNotAvailable is returned. Watch authentication dialog looks and behaves simi ant. Users can confirm authentication by double-c their watch.

17.2 class NSColorPickerTouchBarItemMBS

17.2.1 class NSColorPickerTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar item showing a color picker.
Notes: Subclass of the NSTouchBarItemMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

Xojo Developer Magazine

- 15.5, page 35: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.2.2 Methods

17.2.3 colorPicker(identifier as string) as NSColorPickerTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Creates a bar item containing a button with the standard color picker icon that invokes the color picker.
Example:

```
dim t as NSColorPickerTouchBarItemMBS = NSColorPickerTouchBarItemMBS.colorPicker("test")
t.customizationLabel = "Drawing Color"
```

Notes: You may want to use AddHandler method to add an event handler to this object.
See also:

- 17.2.4 colorPicker(identifier as string, buttonImage as NSImageMBS) as NSColorPickerTouchBarItemMBS 1279

17.2.4 colorPicker(identifier as string, buttonImage as NSImageMBS) as NSColorPickerTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Creates a bar item containing a button with the provided image that invokes the color picker.
Notes: You may want to use AddHandler method to add an event handler to this object.
See also:

- 17.2.3 colorPicker(identifier as string) as NSColorPickerTouchBarItemMBS 1279

17.2.5 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Example:

```
dim t as new NSColorPickerTouchBarItemMBS("test")
t.customizationLabel = "Drawing Color"
```

Notes: This instantiates a new touch bar item with the specified initializer.

17.2.6 strokeColorPicker(identifier as string) as NSColorPickerTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Creates a bar item containing a button with the standard stroke color picker icon that invokes the color picker.
Notes: You may want to use AddHandler method to add an event handler to this object.
Should be used when the item is used for picking stroke colors.

17.2.7 textColorPicker(identifier as string) as NSColorPickerTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Creates a bar item containing a button with the standard text color picker icon that invokes the color picker.
Notes: You may want to use AddHandler method to add an event handler to this object.
Should be used when the item is used for picking text colors.

17.2.8 Properties

17.2.9 color as NSColorMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The selected color of the picker.
Notes: (Read and Write property)

17.2.10 colorList as NSColorListMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The color list displayed in the list color picker.
Notes: Defaults to the standard system color list. Setting a custom color list will disable the additional tints/shades that appear on long-press.
(Read and Write property)

17.2.11 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Example:

```
dim t as new NSColorPickerTouchBarItemMBS("test")
t.customizationLabel = "Drawing Color"
```

Notes: The default value is empty string.
(Read and Write property)

17.2.12 enabled as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Enables or disabled the color picker.
Notes: If it is currently being shown in a popover, it will be dismissed.
(Read and Write property)

17.2.13 showsAlpha as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the picker should allow picking a color with non-1.0 alpha.
Notes: Defaults to "not NSColor.ignoresAlpha".
(Read and Write property)

17.2.14 Events

17.2.15 Action

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The color picker changed color.

17.3 class NSCustomTouchBarItemMBS

17.3.1 class NSCustomTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar item showing a custom view.
Notes: Subclass of the NSTouchBarItemMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

Xojo Developer Magazine

- 15.5, page 35: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.3.2 Methods

17.3.3 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Example:

```
dim t as new NSCustomTouchBarItemMBS("test")
t.customizationLabel = "My Test"
```

Notes: This instantiates a new touch bar item with the specified initializer.

17.3.4 Properties

17.3.5 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Example:

```
dim t as new NSCustomTouchBarItemMBS("test")
t.customizationLabel = "Drawing Color"
```

Notes: The default value is empty string.
(Read and Write property)

17.3.6 view as NSViewMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A view to be displayed in the touch bar in the location corresponding to this item.
Notes: By default, the getter for this property will return this item's view controller's view. If this property is set explicitly, the view controller will be set to nil.
(Read and Write property)

17.3.7 viewController as NSViewControllerMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A view controller whose view is to be displayed in the touch bar in the location corresponding to this item.
Notes: By default, this property is nil.
When set, this item's view property will automatically return the view associated with this view controller.
(Read and Write property)

17.4 class NSGroupTouchBarItemMBS

17.4.1 class NSGroupTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar item showing a group.
Notes: Subclass of the NSTouchBarItemMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

17.4.2 Methods

17.4.3 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Example:

```
dim t as new NSGroupTouchBarItemMBS("test")
t.customizationLabel = "My Group"
```

Notes: This instantiates a new touch bar item with the specified initializer.

17.4.4 groupItemWithIdentifier(identifier as string, items() as NSTouchBarItemMBS) as NSGroupTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Returns an autoreleased NSGroupTouchBarItem with a groupTouchBar built from the given items array.
Example:

```
dim t as NSGroupTouchBarItemMBS = NSGroupTouchBarItemMBS.groupItemWithIdentifier("test")
t.customizationLabel = "My Group"
```

Notes: Customization is not enabled by default when creating a NSGroupTouchBarItem this way.

17.4.5 Properties

17.4.6 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Notes: The default value is empty string.
(Read and Write property)

17.4.7 groupTouchBar as NSViewControllerMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A touch bar, presented seamlessly as part of the touch bar this item is hosted in.
Notes: This touch bar may have its own principal item, and can be customized (or not) per the normal touch bar customization rules.
By default this is an empty touch bar that cannot be customized.
(Read and Write property)

17.5 class NSPopoverTouchBarItemMBS

17.5.1 class NSPopoverTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar item showing a popover.
Notes: Subclass of the NSTouchBarItemMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

Xojo Developer Magazine

- 15.5, page 35: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.5.2 Methods

17.5.3 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Notes: This instantiates a new touch bar item with the specified initializer.

17.5.4 dismissPopover

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: This method will restore the previously visible main touch bar.
Notes: This method can be invoked explicitly to order out a popover if interacting with an item inside it should close it.

17.5.5 showPopover

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Replaces the main touch bar with this item's popover touch bar.
Notes: If this item is not visible, this method will have no effect. If this item ceases to be visible, the popover touch bar will automatically be ordered out.

17.5.6 Properties

17.5.7 collapsedRepresentation as NSViewMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The view displayed when the item is in its hosted touch bar.
Notes: By default, this is an NSButton whose target is this popover item, whose action is showPopover, and whose image and title are bound to this item's collapsedRepresentationImage and collapsedRepresentationLabel respectively.
(Read and Write property)

17.5.8 collapsedRepresentationImage as NSImageMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The image displayed by the button used by default for the default collapsed representation.
Notes: If the collapsedRepresentation button has been replaced by a different view, this property may not have any effect.
(Read and Write property)

17.5.9 collapsedRepresentationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string displayed by the button used by default for the default collapsed representation.
Notes: If the collapsedRepresentation button has been replaced by a different view, this property may not have any effect. This property is archived.
(Read and Write property)

17.5.10 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Notes: The default value is empty string.
(Read and Write property)

17.5.11 popoverTouchBar as NSTouchBarMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The touch bar displayed when this item is "popped."
Notes: By default this is an empty touch bar that cannot be customized. This property is archived.
(Read and Write property)

17.5.12 pressAndHoldTouchBar as NSTouchBarMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A touchbar to be used exclusively for press-and-hold popovers.
Notes: This touch bar can be the same as the one used for "popoverTouchBar" property, but does not have to be. When non-nil this touch bar will be displayed while the user holds their finger down on the collapsed representation and released when the user raises their finger. This tracking behavior is automatic, but popovers with custom collapsed representations will still need to send showPopover to start tracking.
(Read and Write property)

17.5.13 showsCloseButton as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: When true, automatically displays a close button in the popover.
Notes: When false it is the responsibility of the client to dismiss the popover.
(Read and Write property)

17.6 class NSSliderTouchBarItemMBS

17.6.1 class NSSliderTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar item showing a slider.
Notes: Subclass of the NSTouchBarItemMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

Xojo Developer Magazine

- 15.5, page 35: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.6.2 Methods

17.6.3 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.customizationLabel = "My Slider"
```

Notes: This instantiates a new touch bar item with the specified initializer.

17.6.4 Properties

17.6.5 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Notes: The default value is empty string.
(Read and Write property)

17.6.6 label as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The text label displayed along with the slider.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.label = "Pen Size"
```

Notes: If set to nil, the label will not have space reserved in the item.
(Read and Write property)

17.6.7 maxValue as Double

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The maximum value of the slider.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.minValue = 0.0
t.maxValue = 1.0
t.value = 0.5
```

Notes: (Read and Write property)

17.6.8 minValue as Double

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The minimum value of the slider.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.minValue = 0.0
t.maxValue = 1.0
t.value = 0.5
```

Notes: (Read and Write property)

17.6.9 slider as NSSliderMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The slider displayed by the bar item.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.minValue = 0.0
t.maxValue = 1.0
t.value = 0.5
```

Notes: It is automatically created, but can be set to a custom subclass.
Value, minValue, maxValue, etc can all be read and set through the slider.
(Read and Write property)

17.6.10 value as Double

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The current value of the slider.
Notes: (Read and Write property)

17.6.11 Events

17.6.12 Action

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The slider changed.

17.7 class NSTouchBarItemMBS

17.7.1 class NSTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a touch bar item.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.5pr5

Videos

- TouchBar with MBS Xojo Plugin

Xojo Developer Magazine

- 15.5, pages 42 to 44: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar
- 15.5, page 39: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar
- 15.5, page 37: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.7.2 Methods

17.7.3 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Whether this class is available.
Example:

```
if NSTouchBarItemMBS.Available then
MsgBox "This Mac may have a TouchBar or not."
else
MsgBox "This Mac is too old for a TouchBar."
end if
```

Notes: Returns true on macOS 10.12.1 and newer, if the version with TouchBar support is installed.

17.7.4 Constructor(identifier as string)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The designated initializer.
Notes: This instantiates a new touch bar item with the specified initializer.

17.7.5 NSTouchBarItemIdentifierFixedSpaceLarge as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The identifier of an item appropriate for use as a large space in the touch bar.
Example:

```
// create flexible space item
dim t as new NSTouchBarItemMBS(NSTouchBarItemMBS.NSTouchBarItemIdentifierFixedSpaceLarge)
```

Notes: Generally, you can use this identifier in a touch bar's itemIdentifiers array, and it will instantiate that space for you.

17.7.6 NSTouchBarItemIdentifierFixedSpaceSmall as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The identifier of an item appropriate for use as a small space in the touch bar.
Example:

```
// create fixed space item
dim t as new NSTouchBarItemMBS(NSTouchBarItemMBS.NSTouchBarItemIdentifierFixedSpaceSmall)
```

Notes: Generally, you can use this identifier in a touch bar's itemIdentifiers array, and it will instantiate that space for you.

17.7.7 NSTouchBarItemIdentifierFlexibleSpace as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The identifier of an item appropriate for use as a flexible space in the touch bar.
Example:

```
// create flexible space item
dim t as new NSTouchBarItemMBS(NSTouchBarItemMBS.NSTouchBarItemIdentifierFlexibleSpace)
```

Notes: Generally, you can use this identifier in a touch bar's itemIdentifiers array, and it will instantiate that space for you.

17.7.8 NSTouchBarItemIdentifierOtherItemsProxy as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The identifier of the special "other items proxy."
Example:

```
// create proxy item
dim t as new NSTouchBarItemMBS(NSTouchBarItemMBS.NSTouchBarItemIdentifierOtherItemsProxy)
```

Notes: Generally, you can use this identifier in a touch bar's itemIdentifiers array, and a special proxy item will be instantiated for you. When the touch bar containing this item is visible, touch bars provided by items closer to the first responder will be nested inside the space denoted for this item. Space items on either side of this item will be automatically massaged to handle cases where the touch bar containing this identifier is itself the bar closest to the first responder (or closer bars are empty.)
Note that a touch bar lacking this item identifier will be replaced in its entirety by touch bars closer to the first responder.

17.7.9 Properties

17.7.10 customizationLabel as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The localized string labelling this item during user customization.
Notes: The default value is empty string.
(Read only property)

17.7.11 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.
Notes: (Read and Write property)

17.7.12 identifier as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The identifier of this item.
Notes: Apart from spaces, item identifiers should be globally unique.
(Read only property)

17.7.13 view as NSViewMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Intended for subclassing.
Notes: By default, this method returns this item's view controller's view.
(Read only property)

17.7.14 viewController as NSViewControllerMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Intended for subclassing.
Notes: By default, this method returns nil.
(Read only property)

17.7.15 visibilityPriority as Single

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: If there are more items in the touch bar than can be displayed, some will be hidden.
Example:

```
dim t as NSSliderTouchBarItemMBS = new NSSliderTouchBarItemMBS("test")
t.visibilityPriority = t.PriorityLow
```

Notes: Items with high visibility priority will be hidden after items with low visibility priority.
(Read and Write property)

17.7.16 visible as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: When true, this item is attached to a visible touch bar, and is being displayed.
Notes: Note that some types of items are never considered visible, for example spaces, other items proxys, and groups.
This property is key value observable.
(Read only property)

17.7.17 Events

17.7.18 Hidden

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called when the item is hidden.

17.7.19 Shown

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called when the item is shown.

17.7.20 Constants

Visibility Priority

Constant	Value	Description
PriorityHigh	1000	nItems with high visibility priority will be hidden after items with low visibility priority.one
PriorityLow	-1000	nItems with high visibility priority will be hidden after items with low visibility priority.one
PriorityNormal	0	nItems with high visibility priority will be hidden after items with low visibility priority.one

17.8 class NSTouchBarMBS

17.8.1 class NSTouchBarMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The class for a TouchBar on a new MacBook Pro.
Blog Entries

- MBS Xojo Plugins, version 17.3pr1
- MBS Xojo / Real Studio Plugins, version 16.5pr7
- MBS Xojo / Real Studio Plugins, version 16.5pr5

Videos

- TouchBar with MBS Xojo Plugin
- Presentation from London conference about MBS Plugins.

Xojo Developer Magazine

- 15.5, page 41: Touch√©, Using Apple's Touch Bar with Xojo by Marc Zeedar

17.8.2 Methods

17.8.3 AssignToApp

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Assigns the TouchBar to be the main TouchBar for the whole app.

17.8.4 AssignToWindow(window as DesktopWindow)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop only.

Function: Assigns the TouchBar to be the given window.
Notes: The TouchBar on MacBook Pro will show this entries, if the given window is the front window.
See also:

- 17.8.5 AssignToWindow(window as NSWindowMBS) 1299
- 17.8.6 AssignToWindow(window as window) 1299

17.8.5 AssignToWindow(window as NSWindowMBS)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Assigns the TouchBar to be the given window.
Notes: The TouchBar on MacBook Pro will show this entries, if the given window is the front window.
See also:

- 17.8.4 AssignToWindow(window as DesktopWindow) 1298
- 17.8.6 AssignToWindow(window as window) 1299

17.8.6 AssignToWindow(window as window)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Assigns the TouchBar to be the given window.
Notes: The TouchBar on MacBook Pro will show this entries, if the given window is the front window.
See also:

- 17.8.4 AssignToWindow(window as DesktopWindow) 1298
- 17.8.5 AssignToWindow(window as NSWindowMBS) 1299

17.8.7 Available as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Whether this class is available.
Example:

```
if NSTouchBarMBS.Available then
MsgBox "This Mac may have a TouchBar or not."
else
MsgBox "This Mac is too old for a TouchBar."
end if
```

Notes: Returns true on macOS 10.12.1 and newer, if the version with TouchBar support is installed.

17.8.8 Constructor

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The constructor.

17.8.9 customizationAllowedItemIdentifiers as String()

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The items that are presented in the customization palette for the user to add to the touch bar.
Notes: These items will be presented to the user in the order specified in this array.

17.8.10 customizationRequiredItemIdentifiers as String()

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Some items are too important to be removed.
Notes: The corresponding item identifiers should be listed here. During customization the user will be prevented from removing these items from the touch bar.

17.8.11 defaultItemIdentifiers as String()

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: An array of identifiers specifying the items in this touch bar.
Notes: When constructing the instantiated items array, the identifiers in this array will be fed through the itemForIdentifier method.
Item identifiers should be globally unique, excepting NSTouchBarItemIdentifierFixedSpaceSmall, NSTouchBarItemIdentifierFixedSpaceLarge, NSTouchBarItemIdentifierFlexibleSpace, and NSTouchBarItemIdentifierOtherItemsProxy.

This array also corresponds to the item ordering for the receiver in the ‚Äúdefault set‚Äù in the customization palette.

17.8.12 itemForIdentifier(identifier as string) as NSTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Returns an instantiated NSTouchBarItem for the given identifier.
Notes: Items are resolved from the following locations, in order:

- items already in the instantiated items array
- items in the defaultTouchBarItems set
- items returned from the delegate's -touchBar:makeItemForIdentifier: method
- some special identifiers are handled automatically

NSTouchBarItemIdentifierFixedSpaceSmall ->NSTouchBar will automatically create a standard small space
NSTouchBarItemIdentifierFixedSpaceLarge ->NSTouchBar will automatically create a standard large space
NSTouchBarItemIdentifierFlexibleSpace ->NSTouchBar will automatically create a standard flexible space
NSTouchBarItemIdentifierOtherItemsProxy ->NSTouchBar will automatically create a special item that acts as a proxy for the items of touch bars closer to the first responder.

17.8.13 itemIdentifiers as String()

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The resolved array of item identifiers. If the bar has not been customized this will match the defaultItemIdentifiers.

17.8.14 RemoveTouchBarFromApp

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Removes TouchBar for app.

17.8.15 RemoveTouchBarFromWindow(window as NSWindowMBS)

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Removes TouchBar for window.
See also:

- 17.8.16 RemoveTouchBarFromWindow(window as window) 1301

17.8.16 RemoveTouchBarFromWindow(window as window)

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Removes TouchBar for window.
See also:

- 17.8.15 RemoveTouchBarFromWindow(window as NSWindowMBS) 1301

17.8.17 setCustomizationAllowedItemIdentifiers(Identifiers() as String)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Sets the items that are presented in the customization palette for the user to add to the touch bar.
Notes: These items will be presented to the user in the order specified in this array.

17.8.18 setCustomizationRequiredItemIdentifiers(Identifiers() as String)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Sets the items are too important to be removed.
Notes: The corresponding item identifiers should be listed here. During customization the user will be prevented from removing these items from the touch bar.

17.8.19 setDefaultItemIdentifiers(Identifiers() as String)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Sets the array of identifiers specifying the items in this touch bar.

17.8.20 setTemplateItems(Identifiers() as NSTouchBarItemMBS)

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Sets the template item identifiers.
Notes: Items in this set are the first step in resolving instantiated items from their identifiers.
If an item identifier is specified in the itemIdentifiers array, and an item with that identifier is in this set, it will be added to the items array in the corresponding location.

17.8.21 templateItems as NSTouchBarItemMBS()

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Items in this set are the first step in resolving instantiated items from their identifiers.
Notes: If an item identifier is specified in the itemIdentifiers array, and an item with that identifier is in this set, it will be added to the items array in the corresponding location.

17.8.22 toggleTouchBarCustomizationPalette

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Show or dismiss the customization palette for the currently displayed touch bars.
Notes: NSApplication validates this selector against whether the current touch bars are customizable and, if configured on a menu item, will standardize and localize the title. If the current system does not have touch bar support, the menu item will be automatically hidden.

17.8.23 Properties

17.8.24 automaticCustomizeTouchBarMenuItemEnabled as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not a menu item to customize the touch bar can be automatically added to the main menu.
Notes: It will only actually be added when a touch bar hardware or simulator is present. Defaults to false. Setting this property to True is the recommended way to add the customization menu item. But if non-standard placement of the menu item is needed, creating a menu item with an action of toggleTouchBarCustomizationPalette can be used instead.
(Read and Write property)

17.8.25 customizationIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A string uniquely identifying this bar for customization purposes.
Notes: All bars with this identifier will have their items coordinated automatically during customization or instantiation.
Touch bars lacking a customizationIdentifier are not customizable.
(Read and Write property)

17.8.26 Handle as Integer

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.
Notes: (Read and Write property)

17.8.27 principalItemIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Specifying a principal item identifier communicates that the item with that identifier has special significance to this touch bar.
Notes: Currently, that item will be placed in the center of the resolved touch row. Note that multiple visible bars may each specify a principal item identifier - but only one of them can have the request honored. (Read and Write property)

17.8.28 visible as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: When true, the touch bar is attached to an eligible touch bar provider, and its items are displayable, assuming adequate space.
Notes: (Read only property)

17.8.29 Events

17.8.30 DidEnterCustomization

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The customization did start.

17.8.31 DidExitCustomization

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The customization did exit.

17.8.32 Hidden

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called when the bar is hidden.

17.8.33 makeItemForIdentifier(identifier as string) as NSTouchBarItemMBS

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: When constructing the items array, this event will be invoked to construct a touch bar item if that item cannot be found in the defaultItems set.

17.8.34 Shown

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: The event called when the bar is shown.

17.8.35 WillEnterCustomization

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The customization will start soon.

17.8.36 WillExitCustomization

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The customization will exit soon.

Chapter 18

WebKit2

18.1 control DesktopWKWebViewControlMBS

18.1.1 control DesktopWKWebViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A WKWebView object displays interactive web content, such as for an in-app browser.
Example:

```
// print a WKWebViewControlMBS to PDF file:

dim browser as WKWebViewControlMBS // your control showing website
Dim pi As NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo

Dim f As FolderItem = SpecialFolder.Desktop.Child("test.pdf")
pi.SetSaveDestination f

Dim po As NSPrintOperationMBS = browser.printOperation(pi)

po.showsPrintPanel = False
po.showsProgressPanel = False

If po.runOperation Then
f.launch
End If
```

Notes: This is a control to use WebKit in 32-bit and WebKit 2 in 64-bit.
Most of the events provided by Xojo for the control will not work.
But we can add features over time as needed.

Let us know if you need a method from WKWebView or an event from WKNavigationDelegate or WKUIDelegate.

Blog Entries

- News from the MBS Xojo Plugins Version 22.0
- MonkeyBread Software Releases the MBS Xojo Plugins in version 22.0
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0
- Three new controls for iOS in Xojo
- HTMLViewer JavaScript communication for Xojo
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.0
- MBS Xojo Plugins in version 19.0
- HTMLViewer JavaScript communication for Xojo
- Custom JavaScript messages to WKWebViewControlMBS
- MonkeyBread Software Releases the MBS Xojo plug-ins in version 17.0

Videos

- XDC 2020 MBS Plugins Presentation
- Presentation from Munich conference about MBS Plugins.
- Presentation from London conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 18.3, page 43: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes
- 17.5, page 41: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes
- 17.2, page 10: News
- 16.2, page 9: News
- 15.2, page 10: News

18.1.2 Methods

18.1.3 addScriptMessageHandler(Name as String)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Adds a script message handler.
Example:

```
// register once
browser.addScriptMessageHandler "test"
// later use in javascript:
browser.EvaluateJavaScript("window.webkit.messageHandlers.test.postMessage('Hello');")
```

Notes: Name: The name of the message handler.

Adding a script message handler with name name causes the JavaScript function window.webkit.messageHandlers.name.postMessage(messageBody) to be defined in all frames in all web views that use the user content controller.

18.1.4 addUserScript(userScript as WKUserScriptMBS)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Adds an user script.

18.1.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Evaluates JavaScript.
Example:

```
Dim e As NSErrorMBS

// returns array of variants with doubles
Dim v As Variant = DesktopWKWebViewControlMBS1.EvaluateJavaScript("o = [ 1,2,3 ] ;", e)
Dim vd() As Variant = v

// returns array of variants with strings
Dim vv As Variant = DesktopWKWebViewControlMBS1.EvaluateJavaScript("o = [ ""a"",""b"",""d"" ] ;",
e)
```

```
Dim vs() As Variant = vv

// returns dictionary
Dim v3 As Variant = DesktopWKWebViewControlMBS1.EvaluateJavaScript("o = { a:1 } ", e)
Dim dd As Dictionary = v3

Break
```

Notes: Synchronous version which waits for JavaScript to return the value.
Error is set in case of errors.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.1.6 EvaluateJavaScript(JavaScript as String, Tag as String = "") 1310

18.1.6 EvaluateJavaScript(JavaScript as String, Tag as String = "")

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Evaluates JavaScript.
Notes: Calls later JavaScriptEvaluated event with result and passed tag value.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.1.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant 1309

18.1.7 goBack

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to the back item in the back-forward list.
Notes: Sets Navigation property.

18.1.8 goForward

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to the forward item in the back-forward list.
Notes: Sets Navigation property.

18.1.9 goToBackForwardListItem(Item as WKBackForwardListItemMBS)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to an item from the back-forward list and sets it as the current item.
Notes: item: The item to which to navigate. Must be one of the items in the web view's back-forward list.

Sets Navigation property.

18.1.10 LoadData(Data as MemoryBlock, MIMEType as String, textEncodingName as String, baseURL as string = "")

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.LoadData "<p>Hello World</p>", "text/html", "UTF-8", ""
```

Notes: data: The data to use as the contents of the webpage.
MIMEType: The MIME type of the data.
characterEncodingName: The data's character encoding name.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.1.11 LoadFileURL(File as FolderItem, readAccessItem as FolderItem)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested file.
Example:

```
Dim w as DesktopWKWebViewControlMBS // your web viewer
Dim HTMLFile as folderItem = getfolderItem("test.html")
```

```
// pass folderitem parent to allow read to other files in same folder
w.LoadFileURL HTMLFile, HTMLFile.parent
```

Notes: Navigates to the requested file URL on the filesystem.
file: The file URL to which to navigate.
readAccessItem: The file or folder to allow read access to.
If readAccessItem references a single file, only that file may be loaded by WebKit.
If readAccessItem references a directory, files inside that file may be loaded by WebKit.

Sets Navigation property.

18.1.12 LoadHTML(htmlText as String, baseURL as string = "")

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.LoadHTML "<p>Hello World</p>"
```

Notes: htmlText: The string to use as the contents of the webpage.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.1.13 LoadURL(URL as string)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested URL.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

call browser.LoadURL "https://www.mbsplugins.de/xojo"
```

Notes: Sets Navigation property.
Sets Navigation property.

18.1.14 LoadURLRequest(Request as NSURLRequestMBS)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested URL.
Notes: Sets Navigation property.

18.1.15 printOperation(printInfo as NSPrintInfoMBS) as NSPrintOperationMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Creates a print operation for the current web viewer.
Example:

```
dim browser as DesktopWKWebViewControlMBS // your web viewer
Dim pi As NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
Dim po As NSPrintOperationMBS = browser.printOperation(pi)

po.showsPrintPanel = True
po.showsProgressPanel = True

po.runOperationModalForWindow(Self)
```

Notes: Works for WebKit 1.x and 2.x.

For WebKit 2.x may run into endless loop for some websites due to bugs in Apple's WebKit framework.

18.1.16 reload

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Reloads the current page.
Example:

```
Public Sub Reload(browser as DesktopWKWebViewControlMBS, force as Boolean)
If force Then
Call browser.reloadFromOrigin
Else
Call browser.reload
```

```
End If
End Sub
```

Notes: Sets Navigation property.

18.1.17 reloadFromOrigin

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Reloads the current page, performing end-to-end revalidation using cache-validating conditionals if possible.
Example:

```
Public Sub Reload(browser as DesktopWKWebViewControlMBS, force as Boolean)
If force Then
call browser.reloadFromOrigin
Else
call browser.reload
End If
End Sub
```

Notes: Sets Navigation property.

18.1.18 removeAllUserScripts

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Removes all associated user scripts.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.removeAllUserScripts
```

18.1.19 removeScriptMessageHandler(Name as String)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Removes a script message handler.
Notes: name: The name of the message handler to remove.

18.1.20 runOpenPanelWithParametersCompleted(URLs() as NSURLMBS)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop only.

Function: Pass result for a runOpenPanelWithParameters event here.
Notes: Pass nil to indicate cancel was pressed.

18.1.21 setMagnification(magnification as double, pointX as double, pointY as double)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Scales the page content by a specified factor and centers the result on a specified point.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

// zoom to 200% on 100/100
browser.setMagnification(2, 100, 100)
```

Notes: magnification: The factor by which to scale the content.
x/y: The point (in view space) to center magnification on. As usual with Cocoa, bottom is y = 0.

For MacOS 64-bit only.

18.1.22 setUsePrivateBrowsing(value as Boolean)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether to create DesktopWKWebViewControlMBS with private browsing enabled.
Example:

```
DesktopWKWebViewControlMBS.setUsePrivateBrowsing True
```

Notes: Set this property to true before creating the DesktopWKWebViewControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

18.1.23 stopLoading

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Stops loading all resources on the current page.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.stopLoading
```

18.1.24 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Requests a snapshot.
Notes: This is synchronous version which waits for image to be finished.
In case of error the result is nil and error may be set.
See also:

- 18.1.25 takeSnapshot(tag as string = "") 1316

18.1.25 takeSnapshot(tag as string = "")

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Requests a snapshot.
Notes: Calls later takeSnapshotCompleted event.
Tag is passed to event to distinguish various snapshot requests.
See also:

- 18.1.24 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS 1316

18.1.26 userScripts as WKUserScriptMBS()

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The user scripts associated with the user content controller.

18.1.27 Properties

18.1.28 AcceptTabs as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether the control should accept tab keys.
Notes: If true, the plugin will not forward the tab keydown/keyup events to Xojo, because Xojo would do switch to next control.
(Read and Write property)

18.1.29 allowFileAccessFromFileURLs as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Allow file access for file URLs.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.allowFileAccessFromFileURLs = True
```

Notes: Access to files is allowed for some files.
Please check WebKit documentation for details.

Set allowFileAccessFromFileURLs and allowUniversalAccessFromFileURLs to true to disable all the security checks to block local file access for websites. allowUniversalAccessFromFileURLs would allow all file URLs. allowFileAccessFromFileURLs would only allow in same path.
(Read and Write property)

18.1.30 allowsBackForwardNavigationGestures as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether horizontal swipe gestures will trigger back-forward list navigations.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.allowsBackForwardNavigationGestures = True
```

Notes: The default value is false.
Available for 64-bit on macOS 10.10 or newer.
(Read and Write property)

18.1.31 allowsLinkPreview as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines whether pressing on a link displays a preview of the destination for the link.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsLinkPreview = True
```

Notes: (Read and Write property)

18.1.32 allowsMagnification as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether magnify gestures will change the web view,Äôs magnification.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsMagnification = True
```

Notes: The default value is false. You can set the magnification property even if allowsMagnification is set to false.
For MacOS 64-bit only.
(Read and Write property)

18.1.33 allowUniversalAccessFromFileURLs as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Allow universal file access for URLs.
Notes: Any file URL will be loaded if true.

Please check WebKit documentation for details.

Does not work in MacOS 10.14.
(Read and Write property)

18.1.34 backForwardList as WKBackForwardListMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The web view's back-forward list.
Notes: (Read only property)

18.1.35 CanGoBack as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether there is a back item in the back-forward list that can be navigated to.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "CanGoBack: "+If(Browser.CanGoBack, "yes", "no")
```

Notes: (Read only property)

18.1.36 CanGoForward as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether there is a forward item in the back-forward list that can be navigated to.
Example:

```
'Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "CanGoForward: "+If(Browser.CanGoForward, "yes", "no")
```

Notes: (Read only property)

18.1.37 customUserAgent as String

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The custom user agent string.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.customUserAgent = "MyBrowser/1.0"
```

Notes: If no custom user agent string has been set, this is set to "".
Available in 32-bit. Available in 64-bit for macOS 10.11 or newer.
(Read and Write property)

18.1.38 developerExtrasEnabled as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether developer extras are enabled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

// enable context menu entry to show developer extras
browser.developerExtrasEnabled = True
```

Notes: For WebKit 2 (64bit) to enable the context menu to inspect items.
(Read and Write property)

18.1.39 EstimatedProgress as Double

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: An estimate of what fraction of the current navigation has been loaded.
Notes: This value ranges from 0.0 to 1.0 based on the total number of bytes expected to be received, including the main document and all of its potential subresources. After a navigation loading completes, the estimatedProgress remains at 1.0 until a new navigation starts, at which point the estimatedProgress is reset to 0.0.
(Read only property)

18.1.40 hasOnlySecureContent as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether all resources on the page have been loaded through securely encrypted connections.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "hasOnlySecureContent: "+If(Browser.hasOnlySecureContent, "yes", "no")
```

Notes: Only set for 64-bit, always false on 32-bit.
(Read only property)

18.1.41 IsLoading as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the view is currently loading content.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

// load a page
browser.LoadURL "https://www.mbsplugins.de/xojo"

// wait synchronously for website loading
While browser.IsLoading
DelayMBS 0.1
Wend

MsgBox "loaded"
```

Notes: (Read only property)

18.1.42 javaEnabled as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether java is enabled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance
```

```
MsgBox "JavaScript Enabled: "+If(Browser.javaEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.1.43 javaScriptCanOpenWindowsAutomatically as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether javascript can open new windows.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

// disable popup windows
Browser.javaScriptCanOpenWindowsAutomatically = False
```

Notes: (Read and Write property)

18.1.44 javaScriptEnabled as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether javascript is enabled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaScriptEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.1.45 loadsImagesAutomatically as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether to load images automatically.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

// disable loading of images
browser.loadsImagesAutomatically = False#
```

Notes: (Read and Write property)

18.1.46 magnification as Double

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The factor by which the page content is currently scaled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.magnification = 2.0
```

Notes: The default value is 1.0.
For MacOS 64-bit only.
(Read and Write property)

18.1.47 minimumFontSize as Double

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The minimum font size to use.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

browser.minimumFontSize = 20
```

Notes: (Read and Write property)

18.1.48 Navigation as WKNavigationMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The current navigation action.
Notes: (Read only property)

18.1.49 plugInsEnabled as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether plugins are enabled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "plugInsEnabled: "+If(Browser.plugInsEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.1.50 privateBrowsing as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether private browsing is enabled.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox "privateBrowsing: "+If(Browser.privateBrowsing, "yes", "no")
```

Notes: If enabled, no data should be stored for cookies or cache on disk, so it's cleared when web viewer is destroyed (Inkognito mode).
Default is non-private mode for the web viewer.

For WebKit 2.x, it looks like you can't change mode after web viewer was created.
So use UsePrivateBrowsing there.
(Read and Write property)

18.1.51 Title as String

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The page title.

Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox Browser.Title
```

Notes: (Read only property)

18.1.52 URL as String

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The active URL.
Example:

```
Dim browser As DesktopWKWebViewControlMBS // your instance

MsgBox Browser.URL
```

Notes: This is the URL that should be reflected in the user interface.
(Read only property)

18.1.53 UsePrivateBrowsing as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether to create DesktopWKWebViewControlMBS with private browsing enabled.
Example:

```
WVWebViewModuleMBS.UsePrivateBrowsing = True
```

Notes: Set this property to true before creating the DesktopWKWebViewControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

Due to a bug in Xojo 2019r2 and older, please use WVWebViewModuleMBS.UsePrivateBrowsing property instead.
(Read and Write property)

18.1.54 View as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The used view.
Notes: Should be either a WebView (32bit) or WKWebView (64bit).
(Read only property)

18.1.55 WKWebView as WKWebViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The used WKWebViewMBS view.
Notes: Always a WKWebViewMBS in 64-bit, but nil for 32-bit applications.
(Read only property)

18.1.56 Events

18.1.57 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

18.1.58 CreateWebView(URL as String, request as NSURLRequestMBS) as Variant

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Creates a new web view.
Notes: If needed we can add configuration, windowFeatures and navigationAction parameters in future version.

Return a web viewer control or nil.

The web view returned must be created with the specified configuration. WebKit will load the request in the returned web view.

If you do not implement this method or return nil, the web view will cancel the navigation.

18.1.59 decidePolicyForNavigationAction(navigationAction as WKNavigationActionMBS, decisionHandler as WKPolicyForNavigationActionDecisionHandlerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Decides whether to allow or cancel a navigation.
Notes: Please inspect navigationAction to find the information about what action happened.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

If you want to open this in a new window/tab, please call decisionHandler.Cancel method here, take the URL and load it in the new WebView in the new window/tab.

Like the CancelClose event in Xojo, you can call decisionHandler.Cancel method to call it.

18.1.60 decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Decides whether to allow or cancel a navigation after its response is known.
Example:

```
Sub decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS) Handles decidePolicyForNavigationResponse
dim MimeType as string = navigationResponse.response.MIMEType

if MimeType = "application/zip" or _
MimeType = "application/x-tar" or _
MimeType = "application/octet-stream" then
// we download archive files
decisionHandler.Download
else
// allow normal website to load
decisionHandler.Allow

end if
```

End Sub

Notes: navigationResponse: Descriptive information about the navigation response.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

18.1.61 DidClose

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The webviewer closed.
Notes: Notifies your app that the DOM window object's close() method completed successfully.

Your app should remove the web view from the view hierarchy and update the UI as needed, such as by closing the containing browser tab or window.

18.1.62 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

18.1.63 didCommitNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when content starts arriving for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.1.64 didFailNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when an error occurs during a committed main frame navigation.

Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.1.65 didFailProvisionalNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when an error occurs while starting to load data for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.1.66 didFinishNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when a main frame navigation completes.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.
Same as DocumentComplete in HTMLViewer.

18.1.67 DidReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the app to respond to an authentication challenge.
Notes: challenge: The authentication challenge.

AuthChallengeDisposition: The option to use to handle the challenge. For a list of options, see AuthChallenge* constants.
credential: The credential to use for authentication when the disposition parameter contains the value AuthChallengeUseCredential. Specify nil to continue without a credential.

If you don,Äôt implement this method, the web view responds to the authentication challenge with the AuthChallengeRejectProtectionSpace disposition.

18.1.68 didReceiveScriptMessage(Body as Variant, name as String)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when a script message is received from a webpage.
Notes: body: The body of the message. Automatic translated from Javascript types.
name: The name of the message handler to which the message is sent.

18.1.69 didReceiveServerRedirectForProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when a server redirect is received for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.1.70 didStartProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when a main frame navigation starts.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.1.71 downloadDecideDestinationUsingResponse(download as WKDownloadMBS, response as NSURLResponseMBS, suggestedFilename as String, byref destination as FolderItem)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the app to provide a file destination where the system should write the download data.
Notes: download: The download that needs a file destination where the systems should write the download data.
response: A response from the server for an HTTP request, or a synthesized response for a blob download.
suggestedFilename: A string with a filename suggestion to use in creating the file destination.
destination: A file destination where the systems should write the download data.

Set destination to where to store the file. Or set to nil to cancel.

The suggested filename can come from the response or from the web content.

The destination file URL must meet the following requirements:

- It,Äôs a file that doesn,Äôt exist.
- It,Äôs in a directory that exists.
- It,Äôs in a directory that WebKit can write to.

18.1.72 downloadDidFailWithError(download as WKDownloadMBS, error as NSErrorMBS, resumeData as MemoryBlock)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells your app that the download failed, with error information and data you can use to restart the download.
Notes: download: The download that failed.
error: An error describing what caused the download to fail.
resumeData: A data object you use to restart the download.

To restart a failed download, call resumeDownloadFromResumeData with resumeData.

18.1.73 downloadDidFinish(download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: This event tells you that the download finished.
Notes: download: The download that finished.

18.1.74 downloadDidReceiveAuthenticationChallenge(download as WKDownloadMBS, challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the delegate to respond to an authentication challenge.
Notes: download: The download that received the authentication challenge.
challenge: The authentication challenge.

Determine how to respond to the authentication challenge in this method. Then set AuthChallengeDisposition to AuthChallenge* constants and pass the credentials to use.

If you don,Äôt implement this method, the web view responds to the challenge with AuthChallengeReject-ProtectionSpace.

18.1.75 downloadWillPerformHTTPRedirection(download as WKDownloadMBS, response as NSURLResponseMBS, request as NSURLRequestMBS, byref DownloadRedirectPolicy as Integer)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the delegate to respond to the download,Äôs redirect response.
Notes: download: The download that receives the redirect response.
response: The redirect response.
request: The new request the web view sends as a result of the redirect response.

Set DownloadRedirectPolicy to either DownloadRedirectPolicyAllow or DownloadRedirectPolicyCancel.

A download redirect policy that indicates whether to proceed with the redirect.

Determine whether to proceed with the redirect. Then invoke the decisionHandler closure, providing a download redirect policy that indicates whether to proceed with the redirect.

If you don,Äôt implement this event, the web view proceeds with all redirects.

18.1.76 EstimatedProgressChanged(estimatedProgress as double, oldEstimatedProgress as double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The estimated progress for loading the webpage changed.
Notes: This event may be called often, so don't do much here!

18.1.77 FocusLost

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control lost focus.
In older Xojo versions, this event is named LostFocus.
Notes:

This only fires if the control itself lost focus and not a sub control.

18.1.78 FocusReceived

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The control itself got focus.
In older Xojo versions, this event is named GotFocus.
Notes:

This only fires if the control itself got focus and not a sub control.

18.1.79 FrameChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

18.1.80 JavaScriptEvaluated(JavaScript as String, Result as Variant, Error as NSErrorMBS, Tag as String)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: JavaScript was evaluated.
Notes: Error is only set with 64-bit, not with 32-bit.
Tag is passed from EvaluateJavaScript call.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.

18.1.81 MenuBarSelected

Plugin Version: 21.5, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named EnableMenuItems.

18.1.82 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

18.1.83 MouseDrag(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

18.1.84 MouseUp(x as Integer, y as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

18.1.85 navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that a navigation action became a download.
Example:

```
Sub navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WKDownloadMBS) Handles navigationActionDidBecomeDownload
```

```
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationAction: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.1.86 navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that a navigation response became a download.
Example:

```
Sub navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, download as WKDownloadMBS) Handles navigationResponseDidBecomeDownload
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationResponse: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.1.87 runJavaScriptAlertPanel(initiatedByFrame as WKFrameInfoMBS, message as String)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Displays a JavaScript alert panel.
Notes: message: The message to display.

If you do not implement this method, the web view will behave as if the user selected the OK button.

18.1.88 runJavaScriptConfirmPanel(initiatedByFrame as WKFrameInfoMBS, message as String) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Displays a JavaScript confirm panel.
Notes: message: The message to display.

Return true if the user chose OK, false if the user chose Cancel.

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.1.89 runJavaScriptTextInputPanel(initiatedByFrame as WKFrameInfoMBS, prompt as String, defaultText as String) as String

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Displays a JavaScript text input panel.
Notes: message: The message to display.
defaultText: The initial text to display in the text entry field.

Return the entered text if the user chose OK, otherwise "".

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.1.90 runOpenPanelWithParameters(initiatedByFrame as WKFrameInfoMBS, allowsMultipleSelection as Boolean, allowsDirectories as Boolean)

Plugin Version: 22.0, Platform: macOS, Targets: .

Function: Displays a file upload panel.
Notes: Please call runOpenPanelWithParametersCompleted method later with result. Pass the selected URLs if the user chose "OK", otherwise nil.

You can use sheet window for the file picker and later call the runOpenPanelWithParametersCompleted method or call it right in the event if you do it synchronously.

If this method is not implemented, the web view behaves as if the user selected the Cancel button.

Not available on iOS. Implemented for macOS 64-bit applications.

18.1.91 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

18.1.92 takeSnapshotCompleted(image as NSImageMBS, error as NSErrorMBS, tag as string)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Called when snapshot of website is completed.
Notes: If snapshot creation failed, error is set.

18.1.93 TitleChanged(Title as String, oldTitle as string)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The title of the webpage changed.

18.1.94 WebContentProcessDidTerminate

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that the web view,Äôs content process was terminated.
Notes: Web views use a separate process to render and manage web content. WebKit calls this method when the process for the specified web view terminates for any reason.

18.1.95 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

18.1.96 Constants

Authentication Challenge Responses

Constant	Value	Description
AuthChallengeCancelAuthenticationChallenge	2	Cancel the entire request. The provided credential parameter is
AuthChallengePerformDefaultHandling	1	Use the default handling for the challenge as though this delegate not implemented. The provided credential parameter is ignored.
AuthChallengeRejectProtectionSpace	3	Reject this challenge, and call the authentication delegate meth the next authentication protection space. The provided credent is ignored.
AuthChallengeUseCredential	0	Use the specified credential, which may be nil.

Redirect Policy

Constant	Value	Description
DownloadRedirectPolicyAllow	1	Allow a redirect to proceed.
DownloadRedirectPolicyCancel	0	Cancel the redirect action.

18.2 class WKBackForwardListItemMBS

18.2.1 class WKBackForwardListItemMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A WKBackForwardListItem object represents a webpage in the back-forward list of a web view.
Notes: For 32-bit a WebHistoryItem object, for 64-bit a WKBackForwardListItem object.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 20.0pr5

18.2.2 Methods

18.2.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

18.2.4 Properties

18.2.5 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.2.6 initialURL as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The URL of the initial request that created this item.
Notes: (Read only property)

18.2.7 title as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The title of the webpage represented by this item.
Notes: (Read only property)

18.2.8 URL as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The URL of the webpage represented by this item.
Notes: (Read only property)

18.3 class WKBackForwardListMBS

18.3.1 class WKBackForwardListMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A WKBackForwardList object is a list of webpages previously visited in a web view that can be reached by going back or forward.
Notes: For 32-bit a WebBackForwardList object, for 64-bit a WKBackForwardList object.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.0
- MBS Xojo Plugins, version 20.0pr5

18.3.2 Methods

18.3.3 backList as WKBackForwardListItemMBS()

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The portion of the list preceding the current item.
Notes: The items are in the order in which they were originally visited.

18.3.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.
See also:

- 18.3.5 Constructor(WKWebView as WKWebViewMBS) 1341

18.3.5 Constructor(WKWebView as WKWebViewMBS)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Queries back forward list for a web view.
See also:

- 18.3.4 Constructor 1341

18.3.6 forwardList as WKBackForwardListItemMBS()

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The portion of the list following the current item.
Notes: The items are in the order in which they were originally visited.

18.3.7 itemAtIndex(index as Integer) as WKBackForwardListItemMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the item at a specified distance from the current item.
Notes: index: Index of the desired list item relative to the current item:
0 for the current item, -1 for the immediately preceding item, 1 for the immediately following item, and so on.

Returns the item at the specified distance from the current item, or nil if the index parameter exceeds the limits of the list.

18.3.8 Properties

18.3.9 backItem as WKBackForwardListItemMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The item immediately preceding the current item, or nil if there isn't one.
Notes: (Read only property)

18.3.10 currentItem as WKBackForwardListItemMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The current item.
Notes: (Read only property)

18.3.11 forwardItem as WKBackForwardListItemMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The item immediately following the current item, or nil if there isn't one.
Notes: (Read only property)

18.3.12 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.4 class WKDownloadMBS

18.4.1 class WKDownloadMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: An object that represents the download of a web resource.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3

Xojo Developer Magazine

- 20.5, page 10: News

18.4.2 Methods

18.4.3 Cancel

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Cancels the download, and optionally captures data so that you can resume the download later.

18.4.4 Constructor

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

18.4.5 Properties

18.4.6 file as FolderItem

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The target file as folderitem.
Notes: (Read only property)

18.4.7 fileURL as String

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The target file as URL.
Notes: (Read only property)

18.4.8 Handle as Integer

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.4.9 originalRequest as NSURLRequestMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: An object that represents the request that initiated the download.
Notes: (Read only property)

18.4.10 totalBytesExpected as Int64

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Total number of bytes expected to download.
Notes: (Read only property)

18.4.11 totalBytesWritten as Int64

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: Number of bytes written to temp file.
Notes: (Read only property)

18.4.12 webView as WKWebViewMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop & iOS.

Function: The web view where the download initiated.
Notes: (Read only property)

18.5 class WKFrameInfoMBS

18.5.1 class WKFrameInfoMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: An object that contains information about a frame on a webpage.
Notes: An instance of this class is a transient, data-only object; it does not uniquely identify a frame across multiple event calls.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3

Xojo Developer Magazine

- 20.5, page 10: News

18.5.2 Methods

18.5.3 Constructor

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

18.5.4 Properties

18.5.5 Handle as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.5.6 isMainFrame as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the frame is the web site's main frame or a subframe.
Notes: (Read only property)

18.5.7 request as NSURLRequestMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The frame‚Äôs current request.
Notes: (Read only property)

18.5.8 webView as WKWebViewMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The matching web viewer.
Notes: (Read only property)

18.6 class WKHTTPCookieStoreMBS

18.6.1 class WKHTTPCookieStoreMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: An object that manages the HTTP cookies associated with a particular WKWebsiteDataStore.
Example:

```
Dim CookieStore As WKHTTPCookieStoreMBS = WKWebViewControlMBS1.WKWebView.HTTPCookieStore
Dim cookies() As NSHTTPCookieMBS = CookieStore.AllCookies
Dim cookie As NSHTTPCookieMBS = cookies(0)

// now check in debugger
Break
```

Notes: Requires macOS 10.13 or iOS 11.0 or newer.
Blog Entries

- News from the MBS Xojo Plugins Version 20.1
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- MBS Xojo Plugins, version 20.1pr4

Xojo Developer Magazine

- 18.3, page 10: News

18.6.2 Methods

18.6.3 AllCookies as NSHTTPCookieMBS()

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: Fetches all stored cookies.
Example:

```
Dim CookieStore As WKHTTPCookieStoreMBS = WKWebViewControlMBS1.WKWebView.HTTPCookieStore
Dim cookies() As NSHTTPCookieMBS = CookieStore.AllCookies
Dim cookie As NSHTTPCookieMBS = cookies(0)

// now check in debugger
Break
```

18.6.4 Constructor(WKWebView as WKWebViewMBS)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor.
Notes: Creates an object for the http cookie store used by web view.
Adds an observer to the cookie store to call DidChange event.

18.6.5 deleteCookie(cookie as NSHTTPCookieMBS, wait as boolean = true)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: Deletes the specified cookie.
Notes: By default waits for this to complete.
Otherwise this is set asynchronously.

18.6.6 Destructor

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The destructor.

18.6.7 setCookie(cookie as NSHTTPCookieMBS, wait as boolean = true)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: Sets a cookie.
Notes: By default waits for this to complete.
Otherwise this is set asynchronously.

18.6.8 Properties

18.6.9 Handle as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.6.10 Events

18.6.11 DidChange

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: Called when the cookies in the cookie store have changed.

18.7 class WKNavigationActionMBS

18.7.1 class WKNavigationActionMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: An object that contains information about an action that causes navigation to occur.
Notes: Use a WKNavigationAction object to make policy decisions about whether to allow navigation within your app,Äôs web view. You don,Äôt create WKNavigationAction objects directly. Instead, the web view creates them and delivers them to the appropriate delegate objects. Use the methods of your delegate to analyze the action and determine whether to allow the resulting navigation to occur.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins in version 22.3

18.7.2 Methods

18.7.3 Constructor

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

18.7.4 Properties

18.7.5 buttonNumber as Integer

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The number of the mouse button causing the navigation to be requested.
Notes: (Read only property)

18.7.6 Handle as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.7.7 modifierFlags as Integer

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: The modifier keys that were in effect when the navigation was requested.
Notes: (Read only property)

18.7.8 navigationType as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The type of action that triggered the navigation.
Notes: See constants.
(Read only property)

18.7.9 newWindow as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Whether this targets a new window.
Notes: (Read only property)

18.7.10 request as NSURLRequestMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The navigation's request.
Notes: (Read only property)

18.7.11 shouldPerformDownload as Boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A value indicating whether the web content used a download attribute to indicate that this should be downloaded.
Notes: Requires macOS 11.3 or iOS 14.5.

This is only true if WebKit saw a download attribut ein the html link.
(Read only property)

18.7.12 sourceFrame as WKFrameInfoMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The frame requesting the navigation.
Notes: (Read only property)

18.7.13 targetFrame as WKFrameInfoMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The target frame, or nil if this is a new window navigation.
Notes: (Read only property)

18.7.14 Constants

Navigation Types

Constant	Value	Description
NavigationTypeBackForward	2	An item from the back-forward list was requested.
NavigationTypeFormResubmitted	4	A form was resubmitted (for example by going back, going forward, or reloading).
NavigationTypeFormSubmitted	1	A form was submitted.
NavigationTypeLinkActivated	0	A link with an href attribute was activated by the user.
NavigationTypeOther	-1	Navigation is taking place for some other reason.
NavigationTypeReload	3	The webpage was reloaded.

18.8 class WKNavigationMBS

18.8.1 class WKNavigationMBS

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: A WKNavigation object contains information for tracking the loading progress of a webpage.
Notes: A navigation object is returned from the web view load methods and is also passed to the navigation delegate methods to uniquely identify a webpage load from start to finish. It has no method or properties of its own.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 20.3
- MBS Xojo Plugins, version 20.3pr5

18.8.2 Methods

18.8.3 Constructor

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

18.8.4 Properties

18.8.5 Handle as Integer

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: The handle for the navigation action.
Notes: (Read only property)

18.8.6 request as NSURLRequestMBS

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: The request.
Notes: May be nil if not available.
(Read only property)

18.9 class WKNavigationResponseMBS

18.9.1 class WKNavigationResponseMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A WKNavigationResponse object contains information about a navigation response, used for making policy decisions.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 20.0pr5

18.9.2 Methods

18.9.3 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

18.9.4 Properties

18.9.5 canShowMIMEType as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether WebKit can display the response,Äôs MIME type natively.
Notes: (Read only property)

18.9.6 Handle as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.9.7 isForMainFrame as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the frame being navigated is the main frame.
Notes: (Read only property)

18.9.8 response as NSURLResponseMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The frame,Äôs response.
Notes: Allowing a navigation response with a MIME type that can,Äôt be shown will cause the navigation to fail.
(Read only property)

18.10 class WKPolicyForNavigationActionDecisionHandlerMBS

18.10.1 class WKPolicyForNavigationActionDecisionHandlerMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The class for the decision handler to tell webkit what to do with the navigation action.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

18.10.2 Methods

18.10.3 Allow

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Allow the navigation action to continue.

18.10.4 Cancel

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Cancel this navigation action.

18.10.5 Constructor

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

18.10.6 Download

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Change the navigation action to a download.

18.11 class WKPolicyForNavigationResponseDecisionHandlerMBS

18.11.1 class WKPolicyForNavigationResponseDecisionHandlerMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The class for the decision handler to tell webkit what to do with the navigation response.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3

18.11.2 Methods

18.11.3 Allow

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Allow the response to proceed.

18.11.4 Cancel

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Cancel the navigation.

18.11.5 Constructor

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

18.11.6 Download

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Change the navigation response to a download.

18.12 class WKPreferencesMBS

18.12.1 class WKPreferencesMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A WKPreferences object encapsulates the preference settings for a web view.
Notes: The preferences object associated with a web view is specified by its web view configuration.
Available on MacOS 10.10 or newer.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.3
- MBS Xojo Plugins, version 23.3pr6
- MBS Xojo Plugins, version 21.5pr4
- News from the MBS Xojo Plugins Version 21.4
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.4
- News from the MBS Xojo Plugins Version 20.4
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.4
- News from the MBS Xojo Plugins Version 20.1
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1

Xojo Developer Magazine

- 21.5, page 10: News
- 19.6, page 10: News
- 18.6, page 10: News
- 18.3, page 10: News

18.12.2 Methods

18.12.3 Constructor(WKWebView as WKWebViewMBS)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Queries preferences object for a web view.

18.12.4 Properties

18.12.5 crossOriginResourcePolicyEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to enable/disable cross origin resource policy.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences

w.crossOriginResourcePolicyEnabled = False
```

Notes: Disabling some cross policy checks may allow JavaScript to access a website loaded in a frame.

Default is true.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.6 defaultFontSize as Integer

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: The default font size for fonts.
Example:

```
HTMLViewer1.WKWebView.Preferences.defaultFontSize = 10
```

Notes: (Read and Write property)

18.12.7 developerExtrasEnabled as Boolean

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether developer extras are enabled.
Example:

```
HTMLViewer1.WKWebView.Preferences.developerExtrasEnabled = true
```

Notes: Set to true, so you can inspect elements via contextual menus.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.8 ElementFullscreenEnabled as Boolean

Plugin Version: 23.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether a web view can display content full screen.
Notes: The default value for this preference is false.
Requires macOS 12.3 or newer.
(Read and Write property)

18.12.9 FraudulentWebsiteWarningEnabled as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether warnings should be shown for suspected fraudulent content such as phishing or malware.
Notes: The default value is true. This feature is currently available in the following region: China.
(Read and Write property)

18.12.10 fullScreenEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to allow fullscreen mode.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences
w.fullScreenEnabled = true
```

Notes: Default is false.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).

(Read and Write property)

18.12.11 Handle as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.12.12 javaEnabled as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether Java is enabled.
Example:

WKWebViewControlMBS1.WKWebView.Preferences.javaEnabled = False

Notes: The default value is false.
Deprecated for macOS 10.15.
(Read and Write property)

18.12.13 javaScriptCanOpenWindowsAutomatically as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether JavaScript can open windows without user interaction.
Example:

WKWebViewControlMBS1.WKWebView.Preferences.javaScriptCanOpenWindowsAutomatically = False

Notes: The default value is false in iOS and true in macOS.
(Read and Write property)

18.12.14 javaScriptEnabled as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether JavaScript is enabled.
Example:

WKWebViewControlMBS1.WKWebView.Preferences.javaScriptEnabled = False

Notes: The default value is true. Setting this property to false disables JavaScripts that are loaded or executed by the web page. This setting does not affect user scripts. See WKUserScriptMBS class.
(Read and Write property)

18.12.15 loadsImagesAutomatically as Boolean

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether images are loaded automatically.
Example:

HTMLViewer1.WKWebView.Preferences.loadsImagesAutomatically = False

Notes: (Read and Write property)

18.12.16 logsPageMessagesToSystemConsoleEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether page log messages go to system console.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences
w.logsPageMessagesToSystemConsoleEnabled = true
```

Notes: Default is false.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.17 mediaDevicesEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to allow access to media devices in JavaScript.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences
w.mediaDevicesEnabled = true
w.mediaStreamEnabled = true
```

Notes: Default is false.

You may need to enable mediaStreamEnabled and mediaDevicesEnabled together to allow WebRTC. Also you need permissions to access camera and microphone via AVFoundation functions.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.18 mediaStreamEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to allow access to media streaming in JavaScript.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences
w.mediaDevicesEnabled = true
w.mediaStreamEnabled = true
```

Notes: Default is false.

You may need to enable mediaStreamEnabled and mediaDevicesEnabled together to allow WebRTC. Also you need permissions to access camera and microphone via AVFoundation functions.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.19 minimumFontSize as Double

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum font size in points.
Example:

WKWebViewControlMBS1.WKWebView.Preferences.minimumFontSize = 20

Notes: The default value is 0.
(Read and Write property)

18.12.20 plugInsEnabled as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether plug-ins are enabled.
Example:

WKWebViewControlMBS1.WKWebView.Preferences.plugInsEnabled = False

Notes: The default value is false.
Deprecated for macOS 10.15.
(Read and Write property)

18.12.21 shouldPrintBackgrounds as Boolean

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether webkit draws background on printing.
Example:

HTMLViewer1.WKWebView.Preferences.shouldPrintBackgrounds = False

Notes: (Read and Write property)

18.12.22 siteSpecificQuirksModeEnabled as Boolean

Plugin Version: 23.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean that indicates whether to apply site-specific compatibility workarounds.
Notes: The default value for this preference is true.
Requires macOS 12.3 or newer.
(Read and Write property)

18.12.23 standardFontFamily as String

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the standard font family.
Notes: (Read and Write property)

18.12.24 tabFocusesLinks as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: If tabFocusesLinks is true, the tab key will focus links and form controls.
Notes: The Option key temporarily reverses this preference.
(Read and Write property)

18.12.25 telephoneNumberDetectionIsEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether phone number detection in text is enabled.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences

w.telephoneNumberDetectionIsEnabled = True
```

Notes: Default is false.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.12.26 TextInteractionEnabled as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop & iOS.

Function: Whether text interactions are enabled.
Notes: Requires macOS 12.0 or iOS 15.0 or newer.
(Read and Write property)

18.12.27 webSecurityEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to enable/disable web security.
Example:

```
Dim w As WKPreferencesMBS = HTMLViewer1.WKWebViewMBS.Preferences
w.webSecurityEnabled = False
```

Notes: Default is true.

This property is included in the WebKit project, but not documented by Apple, so use with care and be aware that it may break in future (or gets officially documented in a future version of macOS).
(Read and Write property)

18.13 class WKUserScriptMBS

18.13.1 class WKUserScriptMBS

Plugin Version: 19.5, Platform: macOS, Targets: Desktop & iOS.

Function: A WKUserScript object represents a script that can be injected into a webpage.
Notes: Requires MacOS 10.10 or newer.
64-bit target only.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.5
- MBS Xojo Plugins, version 19.5pr8

18.13.2 Methods

18.13.3 Constructor(Source as String, injectionTime as Integer, isForMainFrameOnly as Boolean)

Plugin Version: 19.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an initialized user script that can be added to a user content controller.
Notes: source: The script,Äôs source code.
injectionTime: The time when the script should be injected into the webpage. Either InjectionTimeAtDocumentStart
or InjectionTimeAtDocumentEnd.
isForMainFrameOnly: A Boolean value indicating whether the script should be injected only into the main frame (YES) or into all frames (NO).

18.13.4 Properties

18.13.5 Handle as Integer

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.13.6 injectionTime as Integer

Plugin Version: 19.5, Platform: macOS, Targets: Desktop & iOS.

Function: The time when the script should be injected into the webpage.
Notes: Either InjectionTimeAtDocumentStart or InjectionTimeAtDocumentEnd.
(Read only property)

18.13.7 isForMainFrameOnly as Boolean

Plugin Version: 19.5, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the script should be injected only into the main frame (true) or into all frames (false).
Notes: (Read only property)

18.13.8 Source as String

Plugin Version: 19.5, Platform: macOS, Targets: Desktop & iOS.

Function: The script,Äôs source code.
Notes: (Read only property)

18.13.9 Constants

InjectionTime

Constant	Value	Description
InjectionTimeAtDocumentEnd	1	Inject the script after the document finishes loading, but before other subresources finish loading.
InjectionTimeAtDocumentStart	0	Inject the script after the document element is created, but before any other content is loaded.

18.14 class WKWebViewConfigurationMBS

18.14.1 class WKWebViewConfigurationMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A collection of properties used to initialize a web view.
Notes: Using the WKWebViewConfiguration class, you can determine how soon a webpage is rendered, how media playback is handled, the granularity of items that the user can select, and many other options. WKWebViewConfiguration is only used when a web view is first initialized. You cannot use this class to change the web view's configuration after it has been created.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.3
- MBS Xojo Plugins, version 23.3pr6
- News from the MBS Xojo Plugins Version 20.1
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- MBS Xojo Plugins, version 20.1pr4

18.14.2 Methods

18.14.3 Constructor(WKWebView as WKWebViewMBS)

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates Xojo object referencing configuration object for web viewer.

18.14.4 copy as WKWebViewConfigurationMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a copy of the configuration object.

18.14.5 Properties

18.14.6 allowsAirPlayForMediaPlayback as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether AirPlay is allowed.
Notes: The default value is true.
(Read and Write property)

18.14.7 allowsInlineMediaPlayback as Boolean

Plugin Version: 23.3, Platform: iOS, Targets: iOS only.

Function: A Boolean value that indicates whether HTML5 videos play inline or use the native full-screen controller.
Notes: Set this property to true to play videos inline, or false to use the native full-screen controller. When adding a video element to an HTML document on iPhone, you must also include the playsinline attribute. The default value of this property is false for iPhone and true for iPad.
(Read and Write property)

18.14.8 allowsPictureInPictureMediaPlayback as Boolean

Plugin Version: 23.3, Platform: iOS, Targets: iOS only.

Function: A Boolean value that indicates whether HTML5 videos can play Picture in Picture.
Notes: The default value of this property is true.
(Read and Write property)

18.14.9 applicationNameForUserAgent as String

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The name of the application as used in the user agent string.
Notes: (Read and Write property)

18.14.10 Handle as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The internal object reference.
Notes: (Read only property)

18.14.11 limitsNavigationsToAppBoundDomains as Boolean

Plugin Version: 23.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the web view limits navigation to pages within the app,Äôs domain.
Notes: Requires macOS 11.0 or newer.
(Read and Write property)

18.14.12 mediaTypesRequiringUserActionForPlayback as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Determines which media types require a user gesture to begin playing.
Notes: Use AudiovisualMediaTypeNone to indicate that no user gestures are required to begin playing media.
(Read and Write property)

18.14.13 Preferences as WKPreferencesMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The preference object to be used by the web view.
Notes: (Read and Write property)

18.14.14 suppressesIncrementalRendering as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the web view suppresses content rendering until it is fully loaded into memory.
Notes: The default value is false.
(Read and Write property)

18.14.15 upgradeKnownHostsToHTTPS as Boolean

Plugin Version: 23.3, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the web view should automatically upgrade supported HTTP requests to HTTPS.
Notes: Requires macOS 11.3 or newer.

(Read and Write property)

18.14.16 userInterfaceDirectionPolicy as Integer

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: The directionality of user interface elements.
Notes: The default value is serInterfaceDirectionPolicyContent. For other possible values, see constants. Available on MacOS 10.12 or newer.

The policy used to determine the directionality of user interface elements in a web view.
When UserInterfaceDirectionPolicyContent is specified, the directionality of user interface elements is affected by the dir attribute or the direction CSS property. When UserInterfaceDirectionPolicySystem is specified, the directionality of user interface elements is affected by the direction of the view.
(Read and Write property)

18.14.17 Constants

Audiovisual Media Types

Constant	Value	Description
AudiovisualMediaTypeAll	-1	All media types require a user gesture to begin playing.
AudiovisualMediaTypeAudio	1	Media types containing audio require a user gesture to begin playing.
AudiovisualMediaTypeNone	0	No media types require a user gesture to begin playing.
AudiovisualMediaTypeVideo	2	Media types containing video require a user gesture to begin playing.

User Interface Direction Policy

Constant	Value	Description
UserInterfaceDirectionPolicyContent	0	The directionality follows the CSS/HTML/XHTML specifications. This is default value of userInterfaceDirectionPolicy.
UserInterfaceDirectionPolicySystem	1	The directionality follows the view's userInterfaceLayoutDirection propert

18.15 control WKWebViewControlMBS

18.15.1 control WKWebViewControlMBS

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: A WKWebView object displays interactive web content, such as for an in-app browser.
Example:

```
// print a WKWebViewControlMBS to PDF file:

dim browser as WKWebViewControlMBS // your control showing website
Dim pi As NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo

Dim f As FolderItem = SpecialFolder.Desktop.Child("test.pdf")
pi.SetSaveDestination f

Dim po As NSPrintOperationMBS = browser.printOperation(pi)

po.showsPrintPanel = False
po.showsProgressPanel = False

If po.runOperation Then
f.launch
End If
```

Notes: This is a control to use WebKit in 32-bit and WebKit 2 in 64-bit.
Most of the events provided by Xojo for the control will not work.
But we can add features over time as needed.
Let us know if you need a method from WKWebView or an event from WKNavigationDelegate or WKUIDelegate.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 22.0
- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- Three new controls for iOS in Xojo
- HTMLViewer JavaScript communication for Xojo
- MBS Xojo Plugins in version 19.0
- HTMLViewer JavaScript communication for Xojo
- Custom JavaScript messages to WKWebViewControlMBS

- Enable Web Inspector for WebViewer
- MonkeyBread Software Releases the MBS Xojo plug-ins in version 17.0

Videos

- XDC 2020 MBS Plugins Presentation
- Presentation from Munich conference about MBS Plugins.
- Presentation from London conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 18.3, page 43: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes
- 17.5, page 41: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes
- 17.2, page 10: News
- 16.2, page 9: News
- 15.2, page 10: News

18.15.2 Methods

18.15.3 addScriptMessageHandler(Name as String)

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: Adds a script message handler.
Example:

```
// register once
browser.addScriptMessageHandler "test"
// later use in javascript:
browser.EvaluateJavaScript("window.webkit.messageHandlers.test.postMessage('Hello');")
```

Notes: Name: The name of the message handler.

Adding a script message handler with name name causes the JavaScript function window.webkit.messageHandlers.name.postMessage(messageBody) to be defined in all frames in all web views that use the user content controller.

18.15.4 addUserScript(userScript as WKUserScriptMBS)

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: Adds an user script.

18.15.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Evaluates JavaScript.
Example:

```
Dim e As NSErrorMBS

// returns array of variants with doubles
Dim v As Variant = WKWebViewControlMBS1.EvaluateJavaScript("o = [ 1,2,3 ] ;", e)
Dim vd() As Variant = v

// returns array of variants with strings
Dim vv As Variant = WKWebViewControlMBS1.EvaluateJavaScript("o = [ ""a"",""b"",""d"" ] ;", e)
Dim vs() As Variant = vv

// returns dictionary
Dim v3 As Variant = WKWebViewControlMBS1.EvaluateJavaScript("o = { a:1 } ", e)
Dim dd As Dictionary = v3

Break
```

Notes: Synchronous version which waits for JavaScript to return the value.
Error is set in case of errors.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.15.6 EvaluateJavaScript(JavaScript as String, Tag as String = "") 1377

18.15.6 EvaluateJavaScript(JavaScript as String, Tag as String = "")

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: Evaluates JavaScript.

Notes: Calls later JavaScriptEvaluated event with result and passed tag value.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.15.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant 1377

18.15.7 goBack

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Navigates to the back item in the back-forward list.
Notes: Sets Navigation property.

18.15.8 goForward

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Navigates to the forward item in the back-forward list.
Notes: Sets Navigation property.

18.15.9 goToBackForwardListItem(Item as WKBackForwardListItemMBS)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Navigates to an item from the back-forward list and sets it as the current item.
Notes: item: The item to which to navigate. Must be one of the items in the web view's back-forward list.

Sets Navigation property.

18.15.10 LoadData(Data as MemoryBlock, MIMEType as String, textEncodingName as String, baseURL as string = "")

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewControlMBS // your instance
```

```
browser.LoadData "<p>Hello World</p>", "text/html", "UTF-8", ""
```

Notes: data: The data to use as the contents of the webpage.
MIMEType: The MIME type of the data.
characterEncodingName: The data's character encoding name.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.15.11 LoadFileURL(File as FolderItem, readAccessItem as FolderItem)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested file.
Example:

```
Dim w as WKWebViewControlMBS // your web viewer
Dim HTMLFile as folderItem = getfolderItem("test.html")

// pass folderitem parent to allow read to other files in same folder
w.LoadFileURL HTMLFile, HTMLFile.parent
```

Notes: Navigates to the requested file URL on the filesystem.
file: The file URL to which to navigate.
readAccessItem: The file or folder to allow read access to.
If readAccessItem references a single file, only that file may be loaded by WebKit.
If readAccessItem references a directory, files inside that file may be loaded by WebKit.

Sets Navigation property.

18.15.12 LoadHTML(htmlText as String, baseURL as string = "")

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.LoadHTML "<p>Hello World</p>"
```

Notes: htmlText: The string to use as the contents of the webpage.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.15.13 LoadURL(URL as string)

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested URL.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

call browser.LoadURL "https://www.mbsplugins.de/xojo"
```

Notes: Sets Navigation property.
Sets Navigation property.

18.15.14 LoadURLRequest(Request as NSURLRequestMBS)

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Navigates to a requested URL.
Notes: Sets Navigation property.

18.15.15 printOperation(printInfo as NSPrintInfoMBS) as NSPrintOperationMBS

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: Creates a print operation for the current web viewer.
Example:

```
dim browser as WKWebViewControlMBS // your web viewer
Dim pi As NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
Dim po As NSPrintOperationMBS = browser.printOperation(pi)

po.showsPrintPanel = True
po.showsProgressPanel = True
```

```
po.runOperationModalForWindow(Self)
```

Notes: Works for WebKit 1.x and 2.x.

For WebKit 2.x may run into endless loop for some websites due to bugs in Apple's WebKit framework.

18.15.16 reload

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Reloads the current page.
Example:

```
Public Sub Reload(browser as WKWebViewControlMBS, force as Boolean)
If force Then
Call browser.reloadFromOrigin
Else
Call browser.reload
End If
End Sub
```

Notes: Sets Navigation property.

18.15.17 reloadFromOrigin

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Reloads the current page, performing end-to-end revalidation using cache-validating conditionals if possible.
Example:

```
Public Sub Reload(browser as WKWebViewControlMBS, force as Boolean)
If force Then
call browser.reloadFromOrigin
Else
call browser.reload
End If
End Sub
```

Notes: Sets Navigation property.

18.15.18 removeAllUserScripts

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: Removes all associated user scripts.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.removeAllUserScripts
```

18.15.19 removeScriptMessageHandler(Name as String)

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: Removes a script message handler.
Notes: name: The name of the message handler to remove.

18.15.20 runOpenPanelWithParametersCompleted(URLs() as NSURLMBS)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop only.

Function: Pass result for a runOpenPanelWithParameters event here.
Notes: Pass nil to indicate cancel was pressed.

18.15.21 setMagnification(magnification as double, pointX as double, pointY as double)

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: Scales the page content by a specified factor and centers the result on a specified point.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

// zoom to 200% on 100/100
browser.setMagnification(2, 100, 100)
```

Notes: magnification: The factor by which to scale the content.
x/y: The point (in view space) to center magnification on. As usual with Cocoa, bottom is y = 0.

For MacOS 64-bit only.

18.15.22 setUsePrivateBrowsing(value as Boolean)

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Whether to create WKWebViewControlMBS with private browsing enabled.
Example:

```
WKWebViewControlMBS.setUsePrivateBrowsing True
```

Notes: Set this property to true before creating the WKWebViewControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

18.15.23 stopLoading

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: Stops loading all resources on the current page.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.stopLoading
```

18.15.24 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Requests a snapshot.
Notes: This is synchronous version which waits for image to be finished.
In case of error the result is nil and error may be set.
See also:

18.15.25 takeSnapshot(tag as string = "")

Plugin Version: 17.4, Platform: macOS, Targets: Desktop only.

Function: Requests a snapshot.
Notes: Calls later takeSnapshotCompleted event.
Tag is passed to event to distinguish various snapshot requests.
See also:

18.15.26 userScripts as WKUserScriptMBS()

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: The user scripts associated with the user content controller.

18.15.27 Properties

18.15.28 AcceptTabs as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: Desktop only.

Function: Whether the control should accept tab keys.
Notes: If true, the plugin will not forward the tab keydown/keyup events to Xojo, because Xojo would do switch to next control.
(Read and Write property)

18.15.29 allowFileAccessFromFileURLs as Boolean

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: Allow file access for file URLs.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.allowFileAccessFromFileURLs = True
```

Notes: Access to files is allowed for some files.
Please check WebKit documentation for details.

Set allowFileAccessFromFileURLs and allowUniversalAccessFromFileURLs to true to disable all the security checks to block local file access for websites. allowUniversalAccessFromFileURLs would allow all file URLs. allowFileAccessFromFileURLs would only allow in same path.
(Read and Write property)

18.15.30 allowsBackForwardNavigationGestures as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether horizontal swipe gestures will trigger back-forward list navigations.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.allowsBackForwardNavigationGestures = True
```

Notes: The default value is false.
Available for 64-bit on macOS 10.10 or newer.
(Read and Write property)

18.15.31 allowsLinkPreview as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value that determines whether pressing on a link displays a preview of the destination for the link.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsLinkPreview = True
```

Notes: (Read and Write property)

18.15.32 allowsMagnification as Boolean

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether magnify gestures will change the web view,Äôs magnification.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsMagnification = True
```

Notes: The default value is false. You can set the magnification property even if allowsMagnification is set to false.
For MacOS 64-bit only.
(Read and Write property)

18.15.33 allowUniversalAccessFromFileURLs as Boolean

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: Allow universal file access for URLs.
Notes: Any file URL will be loaded if true.
Please check WebKit documentation for details.

Does not work in MacOS 10.14.
(Read and Write property)

18.15.34 backForwardList as WKBackForwardListMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: The web view's back-forward list.
Notes: (Read only property)

18.15.35 CanGoBack as Boolean

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether there is a back item in the back-forward list that can be navigated to.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "CanGoBack: "+If(Browser.CanGoBack, "yes", "no")
```

Notes: (Read only property)

18.15.36 CanGoForward as Boolean

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether there is a forward item in the back-forward list that can be navigated to.
Example:

```
'Dim browser As WKWebViewControlMBS // your instance

MsgBox "CanGoForward: "+If(Browser.CanGoForward, "yes", "no")
```

Notes: (Read only property)

18.15.37 customUserAgent as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: The custom user agent string.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.customUserAgent = "MyBrowser/1.0"
```

Notes: If no custom user agent string has been set, this is set to "".
Available in 32-bit. Available in 64-bit for macOS 10.11 or newer.
(Read and Write property)

18.15.38 developerExtrasEnabled as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether developer extras are enabled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

// enable context menu entry to show developer extras
browser.developerExtrasEnabled = True
```

Notes: For WebKit 2 (64bit) to enable the context menu to inspect items.
(Read and Write property)

18.15.39 EstimatedProgress as Double

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: An estimate of what fraction of the current navigation has been loaded.
Notes: This value ranges from 0.0 to 1.0 based on the total number of bytes expected to be received, including the main document and all of its potential subresources. After a navigation loading completes, the estimatedProgress remains at 1.0 until a new navigation starts, at which point the estimatedProgress is reset to 0.0.
(Read only property)

18.15.40 hasOnlySecureContent as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether all resources on the page have been loaded through securely encrypted connections.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "hasOnlySecureContent: "+If(Browser.hasOnlySecureContent, "yes", "no")
```

Notes: Only set for 64-bit, always false on 32-bit.
(Read only property)

18.15.41 IsLoading as Boolean

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the view is currently loading content.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

// load a page
browser.LoadURL "https://www.mbsplugins.de/xojo"

// wait synchronously for website loading
While browser.IsLoading
DelayMBS 0.1
Wend

MsgBox "loaded"
```

Notes: (Read only property)

18.15.42 javaEnabled as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether java is enabled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.15.43 javaScriptCanOpenWindowsAutomatically as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether javascript can open new windows.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

// disable popup windows
Browser.javaScriptCanOpenWindowsAutomatically = False
```

Notes: (Read and Write property)

18.15.44 javaScriptEnabled as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether javascript is enabled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaScriptEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.15.45 loadsImagesAutomatically as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether to load images automatically.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

// disable loading of images
browser.loadsImagesAutomatically = False#
```

Notes: (Read and Write property)

18.15.46 magnification as Double

Plugin Version: 19.4, Platform: macOS, Targets: Desktop only.

Function: The factor by which the page content is currently scaled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.magnification = 2.0
```

Notes: The default value is 1.0.
For MacOS 64-bit only.
(Read and Write property)

18.15.47 minimumFontSize as Double

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: The minimum font size to use.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

browser.minimumFontSize = 20
```

Notes: (Read and Write property)

18.15.48 Navigation as WKNavigationMBS

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: The current navigation action.
Notes: (Read only property)

18.15.49 plugInsEnabled as Boolean

Plugin Version: 17.3, Platform: macOS, Targets: Desktop only.

Function: Whether plugins are enabled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "plugInsEnabled: "+If(Browser.plugInsEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.15.50 privateBrowsing as Boolean

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: Whether private browsing is enabled.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox "privateBrowsing: "+If(Browser.privateBrowsing, "yes", "no")
```

Notes: If enabled, no data should be stored for cookies or cache on disk, so it's cleared when web viewer is destroyed (Inkognito mode).
Default is non-private mode for the web viewer.

For WebKit 2.x, it looks like you can't change mode after web viewer was created.
So use UsePrivateBrowsing there.
(Read and Write property)

18.15.51 Title as String

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: The page title.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox Browser.Title
```

Notes: (Read only property)

18.15.52 URL as String

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: The active URL.
Example:

```
Dim browser As WKWebViewControlMBS // your instance

MsgBox Browser.URL
```

Notes: This is the URL that should be reflected in the user interface.
(Read only property)

18.15.53 UsePrivateBrowsing as Boolean

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: Whether to create WKWebViewControlMBS with private browsing enabled.
Example:

```
WVWebViewModuleMBS.UsePrivateBrowsing = True
```

Notes: Set this property to true before creating the WKWebViewControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

Due to a bug in Xojo 2019r2 and older, please use WVWebViewModuleMBS.UsePrivateBrowsing property instead.
(Read and Write property)

18.15.54 View as NSViewMBS

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: The used view.
Notes: Should be either a WebView (32bit) or WKWebView (64bit).
(Read only property)

18.15.55 WKWebView as WKWebViewMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: The used WKWebViewMBS view.
Notes: Always a WKWebViewMBS in 64-bit, but nil for 32-bit applications.
(Read only property)

18.15.56 Events

18.15.57 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

18.15.58 Close

Plugin Version: 16.4, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

18.15.59 ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

18.15.60 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: Called when a menuitem is choosen.
Notes: This allows the control to react on its relevant menu items. Please return true if you handled it or false to give others a chance.

18.15.61 CreateWebView(URL as String, request as NSURLRequestMBS) as Variant

Plugin Version: 20.0, Platform: macOS, Targets: .

Function: Creates a new web view.
Notes: If needed we can add configuration, windowFeatures and navigationAction parameters in future version.

Return a web viewer control or nil.

The web view returned must be created with the specified configuration. WebKit will load the request in the returned web view.

If you do not implement this method or return nil, the web view will cancel the navigation.

18.15.62 decidePolicyForNavigationAction(navigationAction as WKNavigationActionMBS, decisionHandler as WKPolicyForNavigationActionDecisionHandlerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Decides whether to allow or cancel a navigation.
Notes: Please inspect navigationAction to find the information about what action happened.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

If you want to open this in a new window/tab, please call decisionHandler.Cancel method here, take the URL and load it in the new WebView in the new window/tab.

Like the CancelClose event in Xojo, you can call decisionHandler.Cancel method to call it.

18.15.63 decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Decides whether to allow or cancel a navigation after its response is known.
Example:

```
Sub decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS) Handles decidePolicyForNavigationResponse
dim MimeType as string = navigationResponse.response.MIMEType

if MimeType = "application/zip" or _
MimeType = "application/x-tar" or _
MimeType = "application/octet-stream" then
// we download archive files
```

```
decisionHandler.Download
else
// allow normal website to load
decisionHandler.Allow

end if
End Sub
```

Notes: navigationResponse: Descriptive information about the navigation response.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

18.15.64 DidClose

Plugin Version: 20.0, Platform: macOS, Targets: .

Function: The webviewer closed.
Notes: Notifies your app that the DOM window object's close() method completed successfully.

Your app should remove the web view from the view hierarchy and update the UI as needed, such as by closing the containing browser tab or window.

18.15.65 didCloseContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 21.0, Platform: macOS, Targets: .

Function: Did close contextual menu.
Notes: Allows you to restart any animation you may have stopped in the willShowContextualMenu event.

18.15.66 didCommitNavigation(navigation as WKNavigationMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when content starts arriving for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.15.67 didFailNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when an error occurs during a committed main frame navigation.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.15.68 didFailProvisionalNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when an error occurs while starting to load data for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.15.69 didFinishNavigation(navigation as WKNavigationMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when a main frame navigation completes.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.
Same as DocumentComplete in HTMLViewer.

18.15.70 DidReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the app to respond to an authentication challenge.
Notes: challenge: The authentication challenge.

AuthChallengeDisposition: The option to use to handle the challenge. For a list of options, see AuthChallenge* constants.
credential: The credential to use for authentication when the disposition parameter contains the value AuthChallengeUseCredential. Specify nil to continue without a credential.

If you don,Äôt implement this method, the web view responds to the authentication challenge with the AuthChallengeRejectProtectionSpace disposition.

18.15.71 didReceiveScriptMessage(Body as Variant, name as String)

Plugin Version: 18.0, Platform: macOS, Targets: .

Function: Invoked when a script message is received from a webpage.
Notes: body: The body of the message. Automatic translated from Javascript types.
name: The name of the message handler to which the message is sent.

18.15.72 didReceiveServerRedirectForProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when a server redirect is received for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.15.73 didStartProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 17.0, Platform: macOS, Targets: .

Function: Invoked when a main frame navigation starts.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.15.74 downloadDecideDestinationUsingResponse(download as WKDownloadMBS, response as NSURLResponseMBS, suggestedFilename as String, byref destination as FolderItem)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the app to provide a file destination where the system should write the download data.
Notes: download: The download that needs a file destination where the systems should write the download data.
response: A response from the server for an HTTP request, or a synthesized response for a blob download.
suggestedFilename: A string with a filename suggestion to use in creating the file destination.
destination: A file destination where the systems should write the download data.

Set destination to where to store the file. Or set to nil to cancel.

The suggested filename can come from the response or from the web content.
The destination file URL must meet the following requirements:

- It,Äôs a file that doesn,Äôt exist.
- It,Äôs in a directory that exists.
- It,Äôs in a directory that WebKit can write to.

18.15.75 downloadDidFailWithError(download as WKDownloadMBS, error as NSErrorMBS, resumeData as MemoryBlock)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells your app that the download failed, with error information and data you can use to restart the download.
Notes: download: The download that failed.
error: An error describing what caused the download to fail.
resumeData: A data object you use to restart the download.

To restart a failed download, call resumeDownloadFromResumeData with resumeData.

18.15.76 downloadDidFinish(download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: This event tells you that the download finished.
Notes: download: The download that finished.

18.15.77 downloadDidReceiveAuthenticationChallenge(download as WKDownloadMBS, challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the delegate to respond to an authentication challenge.
Notes: download: The download that received the authentication challenge.
challenge: The authentication challenge.

Determine how to respond to the authentication challenge in this method. Then set AuthChallengeDisposition to AuthChallenge* constants and pass the credentials to use.

If you don,Äôt implement this method, the web view responds to the challenge with AuthChallengeRejectProtectionSpace.

18.15.78 downloadWillPerformHTTPRedirection(download as WKDownloadMBS, response as NSURLResponseMBS, request as NSURLRequestMBS, byref DownloadRedirectPolicy as Integer)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Asks the delegate to respond to the download,Äôs redirect response.
Notes: download: The download that receives the redirect response.
response: The redirect response.
request: The new request the web view sends as a result of the redirect response.

Set DownloadRedirectPolicy to either DownloadRedirectPolicyAllow or DownloadRedirectPolicyCancel.

A download redirect policy that indicates whether to proceed with the redirect.

Determine whether to proceed with the redirect. Then invoke the decisionHandler closure, providing a download redirect policy that indicates whether to proceed with the redirect.

If you don,Äôt implement this event, the web view proceeds with all redirects.

18.15.79 EnableMenuItems

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

18.15.80 EstimatedProgressChanged(estimatedProgress as double, oldEstimatedProgress as double)

Plugin Version: 19.0, Platform: macOS, Targets: .

Function: The estimated progress for loading the webpage changed.
Notes: This event may be called often, so don't do much here!

18.15.81 FrameChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the frame changed.
Notes: This event notifies you, that the control changed it's bounding frame, which is position and/or size.

18.15.82 GotFocus

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The control itself got focus.
In Xojo version 2021r3 and newer this event is named FocusReceived.
Notes:

This only fires if the control itself got focus and not a sub control.

18.15.83 JavaScriptEvaluated(JavaScript as String, Result as Variant, Error as NSErrorMBS, Tag as String)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: JavaScript was evaluated.
Notes: Error is only set with 64-bit, not with 32-bit.
Tag is passed from EvaluateJavaScript call.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.

18.15.84 LostFocus

Plugin Version: 17.1, Platform: macOS, Targets: .

Function:

The control lost focus.
In Xojo version 2021r3 and newer this event is named FocusLost.

Notes:

This only fires if the control itself lost focus and not a sub control.

18.15.85 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was pressed inside the control,Äôs region at the location passed in to x, y.
Notes: The coordinates x and y are local to the control, i.e. they represent the position of the mouse click relative to the upper-left corner or the Control.
Return True if you are going to handle the MouseDown. In such a case:

- The Action event, if any, will not execute and the state of the object will not change.
- You will receive the MouseDrag and MouseUp events.

If you return False, the system handles the MouseDown so the above event handlers do not get called.

18.15.86 MouseDrag(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: This event fires continuously after the mouse button was pressed inside the Control.
Notes: Mouse location is local to the control passed in to x, y.
As this event is fired continuously (hundreds of time per second), it is your responsibility to determine if the mouse has really moved.

18.15.87 MouseUp(x as Integer, y as Integer)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The mouse button was released.
Notes: Use the x and y parameters to determine if the mouse button was released within the control's boundaries.

18.15.88 navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that a navigation action became a download.
Example:

```
Sub navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WK-
DownloadMBS) Handles navigationActionDidBecomeDownload
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationAction: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.15.89 navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that a navigation response became a download.
Example:

```
Sub navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, down-
load as WKDownloadMBS) Handles navigationResponseDidBecomeDownload
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationResponse: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.15.90 Open

Plugin Version: 16.4, Platform: macOS, Targets: .

Function:

The control is about to was created and you can initialize it.
In Xojo version 2021r3 and newer this event is named Opening.

18.15.91 runJavaScriptAlertPanel(initiatedByFrame as WKFrameInfoMBS, message as String)

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: Displays a JavaScript alert panel.
Notes: message: The message to display.

If you do not implement this method, the web view will behave as if the user selected the OK button.

18.15.92 runJavaScriptConfirmPanel(initiatedByFrame as WKFrameInfoMBS, message as String) as boolean

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: Displays a JavaScript confirm panel.
Notes: message: The message to display.

Return true if the user chose OK, false if the user chose Cancel.

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.15.93 runJavaScriptTextInputPanel(initiatedByFrame as WKFrameInfoMBS, prompt as String, defaultText as String) as String

Plugin Version: 16.5, Platform: macOS, Targets: .

Function: Displays a JavaScript text input panel.
Notes: message: The message to display.
defaultText: The initial text to display in the text entry field.

Return the entered text if the user chose OK, otherwise "".

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.15.94 runOpenPanelWithParameters(initiatedByFrame as WKFrameInfoMBS, allowsMultipleSelection as Boolean, allowsDirectories as Boolean)

Plugin Version: 22.0, Platform: macOS, Targets: .

Function: Displays a file upload panel.
Notes: Please call runOpenPanelWithParametersCompleted method later with result. Pass the selected URLs if the user chose "OK", otherwise nil.

You can use sheet window for the file picker and later call the runOpenPanelWithParametersCompleted method or call it right in the event if you do it synchronously.

If this method is not implemented, the web view behaves as if the user selected the Cancel button.

Not available on iOS. Implemented for macOS 64-bit applications.

18.15.95 ScaleFactorChanged(NewFactor as Double)

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

18.15.96 takeSnapshotCompleted(image as NSImageMBS, error as NSErrorMBS, tag as string)

Plugin Version: 17.4, Platform: macOS, Targets: .

Function: Called when snapshot of website is completed.
Notes: If snapshot creation failed, error is set.

18.15.97 TitleChanged(Title as String, oldTitle as string)

Plugin Version: 19.0, Platform: macOS, Targets: .

Function: The title of the webpage changed.

18.15.98 WebContentProcessDidTerminate

Plugin Version: 22.3, Platform: macOS, Targets: .

Function: Tells the app that the web view,Äôs content process was terminated.
Notes: Web views use a separate process to render and manage web content. WebKit calls this method when the process for the specified web view terminates for any reason.

18.15.99 willShowContextualMenu(menu as NSMenuMBS, NSEvent as NSEventMBS)

Plugin Version: 21.0, Platform: macOS, Targets: .

Function: Will show contextual menu.
Notes: Your chance to modify the menu before it is shown, e.g. to add menu entries.

18.15.100 Constants

Authentication Challenge Responses

Constant	Value	Description
AuthChallengeCancelAuthenticationChallenge	2	Cancel the entire request. The provided credential parameter is
AuthChallengePerformDefaultHandling	1	Use the default handling for the challenge as though this delegate not implemented. The provided credential parameter is ignored.
AuthChallengeRejectProtectionSpace	3	Reject this challenge, and call the authentication delegate meth the next authentication protection space. The provided credent is ignored.
AuthChallengeUseCredential	0	Use the specified credential, which may be nil.

Redirect Policy

Constant	Value	Description
DownloadRedirectPolicyAllow	1	Allow a redirect to proceed.
DownloadRedirectPolicyCancel	0	Cancel the redirect action.

18.16 control WKWebViewIOSControlMBS

18.16.1 control WKWebViewIOSControlMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A WKWebView object displays interactive web content, such as for an in-app browser.
Notes: This is a control is for iOS only and uses 64-bit.

But we can add features over time as needed.
Let us know if you need a method from WKWebView or an event from WKNavigationDelegate or WKUIDelegate.
Blog Entries

- MBS Xojo Plugins, version 22.4pr1
- MBS Xojo Plugins, version 22.1pr6
- MBS Xojo Plugins Version 21.0 News
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0
- Three new controls for iOS in Xojo
- MBS Xojo Plugins, version 20.6pr3

18.16.2 Methods

18.16.3 addScriptMessageHandler(Name as String)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Adds a script message handler.
Example:

```
// register once
browser.addScriptMessageHandler "test"
// later use in javascript:
browser.EvaluateJavaScript("window.webkit.messageHandlers.test.postMessage('Hello');")
```

Notes: Name: The name of the message handler.

Adding a script message handler with name name causes the JavaScript function window.webkit.messageHandlers.name.postMessage(messageBody) to be defined in all frames in all web views that use the user content controller.

18.16.4 addUserScript(userScript as WKUserScriptMBS)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Adds an user script.

18.16.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Evaluates JavaScript.
Example:

```
Dim e As NSErrorMBS

// returns array of variants with doubles
Dim v As Variant = WKWebViewIOSControlMBS1.EvaluateJavaScript("o = [ 1,2,3 ] ;", e)
Dim vd() As Variant = v

// returns array of variants with strings
Dim vv As Variant = WKWebViewIOSControlMBS1.EvaluateJavaScript("o = [ ""a"",""b"",""d"" ] ;", e)
Dim vs() As Variant = vv

// returns dictionary
Dim v3 As Variant = WKWebViewIOSControlMBS1.EvaluateJavaScript("o = { a:1 } ", e)
Dim dd As Dictionary = v3

Break
```

Notes: Synchronous version which waits for JavaScript to return the value.
Error is set in case of errors.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.16.6 EvaluateJavaScript(JavaScript as String, Tag as String = "") 1408

18.16.6 EvaluateJavaScript(JavaScript as String, Tag as String = "")

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Evaluates JavaScript.

Notes: Calls later JavaScriptEvaluated event with result and passed tag value.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.
See also:

- 18.16.5 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant 1408

18.16.7 goBack

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to the back item in the back-forward list.
Notes: Sets Navigation property.

18.16.8 goForward

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to the forward item in the back-forward list.
Notes: Sets Navigation property.

18.16.9 goToBackForwardListItem(Item as WKBackForwardListItemMBS)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to an item from the back-forward list and sets it as the current item.
Notes: item: The item to which to navigate. Must be one of the items in the web view's back-forward list.

Sets Navigation property.

18.16.10 LoadData(Data as MemoryBlock, MIMEType as String, textEncodingName as String, baseURL as string = "")

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance
```

```
browser.LoadData "<p>Hello World</p>", "text/html", "UTF-8", ""
```

Notes: data: The data to use as the contents of the webpage.
MIMEType: The MIME type of the data.
characterEncodingName: The data's character encoding name.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.16.11 LoadFileURL(File as FolderItem, readAccessItem as FolderItem)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to a requested file.
Notes: Navigates to the requested file URL on the filesystem.
file: The file URL to which to navigate.
readAccessItem: The file or folder to allow read access to.
If readAccessItem references a single file, only that file may be loaded by WebKit.
If readAccessItem references a directory, files inside that file may be loaded by WebKit.

Sets Navigation property.

18.16.12 LoadHTML(htmlText as String, baseURL as string = "")

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.LoadHTML "<p>Hello World</p>"
```

Notes: htmlText: The string to use as the contents of the webpage.
baseURL: A URL used to resolve relative URLs within the document.

Sets Navigation property.

18.16.13 LoadURL(URL as string)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to a requested URL.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

call browser.LoadURL "https://www.mbsplugins.de/xojo"
```

Notes: Sets Navigation property.
Sets Navigation property.

18.16.14 LoadURLRequest(Request as NSURLRequestMBS)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Navigates to a requested URL.
Notes: Sets Navigation property.

18.16.15 reload

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Reloads the current page.
Example:

```
Public Sub Reload(browser as WKWebViewIOSControlMBS, force as Boolean)
If force Then
Call browser.reloadFromOrigin
Else
Call browser.reload
End If
End Sub
```

Notes: Sets Navigation property.

18.16.16 reloadFromOrigin

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Reloads the current page, performing end-to-end revalidation using cache-validating conditionals if possible.
Example:

```
Public Sub Reload(browser as WKWebViewIOSControlMBS, force as Boolean)
If force Then
call browser.reloadFromOrigin
Else
call browser.reload
End If
End Sub
```

Notes: Sets Navigation property.

18.16.17 removeAllUserScripts

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Removes all associated user scripts.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.removeAllUserScripts
```

18.16.18 removeScriptMessageHandler(Name as String)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Removes a script message handler.
Notes: name: The name of the message handler to remove.

18.16.19 setUsePrivateBrowsing(value as Boolean)

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether to create WKWebViewIOSControlMBS with private browsing enabled.
Example:

```
WKWebViewIOSControlMBS.setUsePrivateBrowsing True
```

Notes: Set this property to true before creating the WKWebViewIOSControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

18.16.20 stopLoading

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Stops loading all resources on the current page.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.stopLoading
```

18.16.21 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Requests a snapshot.
Notes: This is synchronous version which waits for image to be finished.
In case of error the result is nil and error may be set.
See also:

- 18.16.22 takeSnapshot(tag as string = "") 1413

18.16.22 takeSnapshot(tag as string = "")

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Requests a snapshot.
Notes: Calls later takeSnapshotCompleted event.
Tag is passed to event to distinguish various snapshot requests.
See also:

- 18.16.21 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS 1413

18.16.23 userScripts as WKUserScriptMBS()

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The user scripts associated with the user content controller.

18.16.24 Properties

18.16.25 allowFileAccessFromFileURLs as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Allow file access for file URLs.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.allowFileAccessFromFileURLs = True
```

Notes: Access to files is allowed for some files.
Please check WebKit documentation for details.

Set allowFileAccessFromFileURLs and allowUniversalAccessFromFileURLs to true to disable all the security checks to block local file access for websites. allowUniversalAccessFromFileURLs would allow all file URLs. allowFileAccessFromFileURLs would only allow in same path.
(Read and Write property)

18.16.26 allowsBackForwardNavigationGestures as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value indicating whether horizontal swipe gestures will trigger back-forward list navigations.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.allowsBackForwardNavigationGestures = True
```

Notes: The default value is false.
Available for 64-bit on macOS 10.10 or newer.
(Read and Write property)

18.16.27 allowsLinkPreview as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value that determines whether pressing on a link displays a preview of the destination for the link.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsLinkPreview = True
```

Notes: (Read and Write property)

18.16.28 allowUniversalAccessFromFileURLs as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Allow universal file access for URLs.
Notes: Any file URL will be loaded if true.
Please check WebKit documentation for details.

Does not work in MacOS 10.14.
(Read and Write property)

18.16.29 backForwardList as WKBackForwardListMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The web view's back-forward list.
Notes: (Read only property)

18.16.30 CanGoBack as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value indicating whether there is a back item in the back-forward list that can be navigated to.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance
```

```
MsgBox "CanGoBack: "+If(Browser.CanGoBack, "yes", "no")
```

Notes: (Read only property)

18.16.31 CanGoForward as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value indicating whether there is a forward item in the back-forward list that can be navigated to.
Example:

```
'Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox "CanGoForward: "+If(Browser.CanGoForward, "yes", "no")
```

Notes: (Read only property)

18.16.32 customUserAgent as String

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The custom user agent string.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.customUserAgent = "MyBrowser/1.0"
```

Notes: If no custom user agent string has been set, this is set to "".
Available in 32-bit. Available in 64-bit for macOS 10.11 or newer.
(Read and Write property)

18.16.33 developerExtrasEnabled as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether developer extras are enabled.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

// enable context menu entry to show developer extras
browser.developerExtrasEnabled = True
```

Notes: For WebKit 2 (64bit) to enable the context menu to inspect items.
(Read and Write property)

18.16.34 EstimatedProgress as Double

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: An estimate of what fraction of the current navigation has been loaded.
Notes: This value ranges from 0.0 to 1.0 based on the total number of bytes expected to be received, including the main document and all of its potential subresources. After a navigation loading completes, the estimatedProgress remains at 1.0 until a new navigation starts, at which point the estimatedProgress is reset to 0.0.
(Read only property)

18.16.35 hasOnlySecureContent as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value indicating whether all resources on the page have been loaded through securely encrypted connections.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox "hasOnlySecureContent: "+If(Browser.hasOnlySecureContent, "yes", "no")
```

Notes: Only set for 64-bit, always false on 32-bit.
(Read only property)

18.16.36 IsLoading as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: A Boolean value indicating whether the view is currently loading content.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

// load a page
browser.LoadURL "https://www.mbsplugins.de/xojo"

// wait synchronously for website loading
While browser.IsLoading
DelayMBS 0.1
Wend

MsgBox "loaded"
```

Notes: (Read only property)

18.16.37 javaScriptCanOpenWindowsAutomatically as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether javascript can open new windows.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

// disable popup windows
Browser.javaScriptCanOpenWindowsAutomatically = False
```

Notes: (Read and Write property)

18.16.38 javaScriptEnabled as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether javascript is enabled.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaScriptEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.16.39 loadsImagesAutomatically as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether to load images automatically.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

// disable loading of images
browser.loadsImagesAutomatically = False#
```

Notes: (Read and Write property)

18.16.40 minimumFontSize as Double

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The minimum font size to use.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

browser.minimumFontSize = 20
```

Notes: (Read and Write property)

18.16.41 Navigation as WKNavigationMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The current navigation action.
Notes: (Read only property)

18.16.42 privateBrowsing as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether private browsing is enabled.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox "privateBrowsing: "+If(Browser.privateBrowsing, "yes", "no")
```

Notes: If enabled, no data should be stored for cookies or cache on disk, so it's cleared when web viewer is destroyed.
Default is non-private mode for the web viewer.

For WebKit 2.x, it looks like you can't change mode after web viewer was created.
So use UsePrivateBrowsing there.
(Read and Write property)

18.16.43 Title as String

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The page title.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox Browser.Title
```

Notes: (Read only property)

18.16.44 URL as String

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The active URL.
Example:

```
Dim browser As WKWebViewIOSControlMBS // your instance

MsgBox Browser.URL
```

Notes: This is the URL that should be reflected in the user interface.
(Read only property)

18.16.45 UsePrivateBrowsing as Boolean

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: Whether to create WKWebViewIOSControlMBS with private browsing enabled.
Example:

WVWebViewModuleMBS.UsePrivateBrowsing = True

Notes: Set this property to true before creating the WKWebViewIOSControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.

Due to a bug in Xojo 2019r2 and older, please use WVWebViewModuleMBS.UsePrivateBrowsing property instead.
(Read and Write property)

18.16.46 View as NSViewMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The used view.
Notes: Should be either a WKWebView (64bit).
(Read only property)

18.16.47 WKWebView as WKWebViewMBS

Plugin Version: 21.0, Platform: iOS, Targets: iOS only.

Function: The used WKWebViewMBS view.
Notes: Always a WKWebViewMBS in 64-bit, but nil for 32-bit applications.
(Read only property)

18.16.48 Events

18.16.49 Close

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control is about to close.

18.16.50 CreateWebView(URL as String, request as NSURLRequestMBS) as WKWebViewIOSControlMBS

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Creates a new web view.
Notes: If needed we can add configuration, windowFeatures and navigationAction parameters in future version.

Return a web viewer control or nil.

The web view returned must be created with the specified configuration. WebKit will load the request in the returned web view.

If you do not implement this method or return nil, the web view will cancel the navigation.

18.16.51 decidePolicyForNavigationAction(navigationAction as WKNavigationActionMBS, decisionHandler as WKPolicyForNavigationActionDecisionHandlerMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Decides whether to allow or cancel a navigation.
Notes: Please inspect navigationAction to find the information about what action happened.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

If you want to open this in a new window/tab, please call decisionHandler.Cancel method here, take the URL and load it in the new WebView in the new window/tab.

Like the CancelClose event in Xojo, you can call decisionHandler.Cancel method to call it.

18.16.52 decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Decides whether to allow or cancel a navigation after its response is known.
Example:

```
Sub decidePolicyForNavigationResponse(navigationResponse as WKNavigationResponseMBS, decisionHandler as WKPolicyForNavigationResponseDecisionHandlerMBS) Handles decidePolicyForNavigationResponse
dim MimeType as string = navigationResponse.response.MIMEType

if MimeType = "application/zip" or _
MimeType = "application/x-tar" or _
MimeType = "application/octet-stream" then
// we download archive files
decisionHandler.Download
else
// allow normal website to load
decisionHandler.Allow

end if
End Sub
```

Notes: navigationResponse: Descriptive information about the navigation response.

You can show a dialog to ask user and once you know the decision call a method on decisionHandler object to report the decision.

18.16.53 DidClose

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The webviewer closed.
Notes: Notifies your app that the DOM window object's close() method completed successfully.

Your app should remove the web view from the view hierarchy and update the UI as needed, such as by closing the containing browser tab or window.

18.16.54 didCommitNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when content starts arriving for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.16.55 didFailNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when an error occurs during a committed main frame navigation.
Notes: Sets Navigation property.

18.16.56 didFailProvisionalNavigation(navigation as WKNavigationMBS, Error as NSErrorMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when an error occurs while starting to load data for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.16.57 didFinishNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when a main frame navigation completes.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.
Same as DocumentComplete in HTMLViewer.

18.16.58 DidReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Asks the app to respond to an authentication challenge.
Notes: challenge: The authentication challenge.

AuthChallengeDisposition: The option to use to handle the challenge. For a list of options, see AuthChallenge* constants.
credential: The credential to use for authentication when the disposition parameter contains the value AuthChallengeUseCredential. Specify nil to continue without a credential.

If you don,Äôt implement this method, the web view responds to the authentication challenge with the AuthChallengeRejectProtectionSpace disposition.

18.16.59 didReceiveScriptMessage(Body as Variant, name as String)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when a script message is received from a webpage.
Notes: body: The body of the message. Automatic translated from Javascript types.
name: The name of the message handler to which the message is sent.

18.16.60 didReceiveServerRedirectForProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when a server redirect is received for the main frame.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.16.61 didStartProvisionalNavigation(navigation as WKNavigationMBS)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Invoked when a main frame navigation starts.
Notes: Added navigation parameter with version 20.3.
Sets Navigation property.

18.16.62 downloadDecideDestinationUsingResponse(download as WKDownloadMBS, response as NSURLResponseMBS, suggestedFilename as String, byref destination as FolderItem)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Asks the app to provide a file destination where the system should write the download data.

Notes: download: The download that needs a file destination where the systems should write the download data.
response: A response from the server for an HTTP request, or a synthesized response for a blob download.
suggestedFilename: A string with a filename suggestion to use in creating the file destination.
destination: A file destination where the systems should write the download data.

Set destination to where to store the file. Or set to nil to cancel.

The suggested filename can come from the response or from the web content.
The destination file URL must meet the following requirements:

- It,Äôs a file that doesn,Äôt exist.
- It,Äôs in a directory that exists.
- It,Äôs in a directory that WebKit can write to.

18.16.63 downloadDidFailWithError(download as WKDownloadMBS, error as NSErrorMBS, resumeData as MemoryBlock)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Tells your app that the download failed, with error information and data you can use to restart the download.
Notes: download: The download that failed.
error: An error describing what caused the download to fail.
resumeData: A data object you use to restart the download.

To restart a failed download, call resumeDownloadFromResumeData with resumeData.

18.16.64 downloadDidFinish(download as WKDownloadMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: This event tells you that the download finished.
Notes: download: The download that finished.

18.16.65 downloadDidReceiveAuthenticationChallenge(download as WKDownloadMBS, challenge as NSURLAuthenticationChallengeMBS, byref AuthChallengeDisposition as Integer, byref credentials as NSURLCredentialMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Asks the delegate to respond to an authentication challenge.
Notes: download: The download that received the authentication challenge.
challenge: The authentication challenge.

Determine how to respond to the authentication challenge in this method. Then set AuthChallengeDisposition to AuthChallenge* constants and pass the credentials to use.

If you don,Äôt implement this method, the web view responds to the challenge with AuthChallengeRejectProtectionSpace.

18.16.66 downloadWillPerformHTTPRedirection(download as WKDownloadMBS, response as NSURLResponseMBS, request as NSURLRequestMBS, byref DownloadRedirectPolicy as Integer)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Asks the delegate to respond to the download,Äôs redirect response.
Notes: download: The download that receives the redirect response.
response: The redirect response.
request: The new request the web view sends as a result of the redirect response.

Set DownloadRedirectPolicy to either DownloadRedirectPolicyAllow or DownloadRedirectPolicyCancel.

A download redirect policy that indicates whether to proceed with the redirect.

Determine whether to proceed with the redirect. Then invoke the decisionHandler closure, providing a download redirect policy that indicates whether to proceed with the redirect.

If you don,Äôt implement this event, the web view proceeds with all redirects.

18.16.67 EstimatedProgressChanged(estimatedProgress as double, oldEstimatedProgress as double)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The estimated progress for loading the webpage changed.
Notes: This event may be called often, so don't do much here!

18.16.68 GotFocus

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control itself got focus.
Notes: This only fires if the control itself got focus and not a sub control.

18.16.69 JavaScriptEvaluated(JavaScript as String, Result as Variant, Error as NSErrorMBS, Tag as String)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: JavaScript was evaluated.
Notes: Error is only set with 64-bit, not with 32-bit.
Tag is passed from EvaluateJavaScript call.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.

18.16.70 LostFocus

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control lost focus.
Notes: This only fires if the control itself lost focus and not a sub control.

18.16.71 navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Tells the app that a navigation action became a download.
Example:

```
Sub navigationActionDidBecomeDownload(navigationAction as WKNavigationActionMBS, download as WK-
DownloadMBS) Handles navigationActionDidBecomeDownload
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationAction: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.16.72 navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, download as WKDownloadMBS)

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Tells the app that a navigation response became a download.
Example:

```
Sub navigationResponseDidBecomeDownload(navigationResponse as WKNavigationResponseMBS, down-
load as WKDownloadMBS) Handles navigationResponseDidBecomeDownload
System.DebugLog CurrentMethodName

// add download to your list of downloads you monitor
End Sub
```

Notes: navigationResponse: Descriptive information about the navigation response that turned into a download.
download: An object that represents the download of a web resource.

Implement this method to begin tracking download progress.

18.16.73 Open

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The control is about to was created and you can initialize it.

18.16.74 runJavaScriptAlertPanel(initiatedByFrame as WKFrameInfoMBS, message as String)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Displays a JavaScript alert panel.
Notes: message: The message to display.

If you do not implement this method, the web view will behave as if the user selected the OK button.

18.16.75 runJavaScriptConfirmPanel(initiatedByFrame as WKFrameInfoMBS, message as String) as boolean

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Displays a JavaScript confirm panel.
Notes: message: The message to display.

Return true if the user chose OK, false if the user chose Cancel.

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.16.76 runJavaScriptTextInputPanel(initiatedByFrame as WKFrameInfoMBS, prompt as String, defaultText as String) as String

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Displays a JavaScript text input panel.
Notes: message: The message to display.
defaultText: The initial text to display in the text entry field.

Return the entered text if the user chose OK, otherwise "".

If you do not implement this method, the web view will behave as if the user selected the Cancel button.

18.16.77 takeSnapshotCompleted(image as NSImageMBS, error as NSErrorMBS, tag as string)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: Called when snapshot of website is completed.
Notes: If snapshot creation failed, error is set.

18.16.78 TitleChanged(Title as String, oldTitle as string)

Plugin Version: 21.0, Platform: iOS, Targets: .

Function: The title of the webpage changed.

18.16.79 WebContentProcessDidTerminate

Plugin Version: 22.3, Platform: iOS, Targets: .

Function: Tells the app that the web view,Äôs content process was terminated.
Notes: Web views use a separate process to render and manage web content. WebKit calls this method when the process for the specified web view terminates for any reason.

18.16.80 Constants

Authentication Challenge Responses

Constant	Value	Description
AuthChallengeCancelAuthenticationChallenge	2	Cancel the entire request. The provided credential parameter is
AuthChallengePerformDefaultHandling	1	Use the default handling for the challenge as though this delegate not implemented. The provided credential parameter is ignored.
AuthChallengeRejectProtectionSpace	3	Reject this challenge, and call the authentication delegate meth the next authentication protection space. The provided credent is ignored.
AuthChallengeUseCredential	0	Use the specified credential, which may be nil.

Redirect Policy

Constant	Value	Description
DownloadRedirectPolicyAllow	1	Allow a redirect to proceed.
DownloadRedirectPolicyCancel	0	Cancel the redirect action.

18.17 class WKWebViewMBS

18.17.1 class WKWebViewMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: An object that displays interactive web content, such as for an in-app browser.
Notes: see
https://developer.apple.com/documentation/webkit/wkwebview

If something is missing, please let us know and we can add it for you.
Subclass of the NSViewMBS class.
Blog Entries

- MBS Xojo Plugins, version 20.5pr7
- News from the MBS Xojo Plugins Version 20.4
- Tip fo the day: Print to PDF for WebView
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.4
- Two weeks with Xojo 2020r1
- Xojo 2020r1 available
- News from the MBS Xojo Plugins Version 20.3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.3
- New in the MBS Xojo Plugins 20.0
- Upgrading our HTMLViewer functions for Internet Explorer

Xojo Developer Magazine

- 18.6, page 10: News

18.17.2 Methods

18.17.3 addUserScript(userScript as WKUserScriptMBS)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Adds an user script.

18.17.4 Constructor

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new web view with size 100/100 and position 0/0
Example:

```
dim x as new WebViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 18.17.5 Constructor(Handle as Integer) 1433
- 18.17.6 Constructor(left as double, top as double, width as double, height as double) 1433

18.17.5 Constructor(Handle as Integer)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an object based on the given WKWebView handle.
Example:

```
dim t as new WKWebViewMBS(0, 0, 100, 100)
dim v as new WKWebViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a WKWebView and the plugin retains this handle.
See also:

- 18.17.4 Constructor 1433
- 18.17.6 Constructor(left as double, top as double, width as double, height as double) 1433

18.17.6 Constructor(left as double, top as double, width as double, height as double)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new button with the given size and position.
Example:

```
dim w as new WKWebViewMBS(0,0,800,600)
```

```
w.LoadHTMLString "Hello World",""
```

Notes: On success the handle property is not zero.
See also:

- 18.17.4 Constructor 1433
- 18.17.5 Constructor(Handle as Integer) 1433

18.17.7 copy

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Copies current text selection.

18.17.8 cut

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Cuts current text selection.

18.17.9 deleteSelection

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Deletes current text selection.

18.17.10 EvaluateJavaScript(JavaScript as String, byref Error as NSErrorMBS) as Variant

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Evaluates JavaScript.
Example:

```
Dim w As WKWebViewMBS = WKWebViewControlMBS1.WKWebView
Dim e As NSErrorMBS

// returns array of variants with doubles
Dim v As Variant = w.EvaluateJavaScript("o = [ 1,2,3 ] ;", e)
```

```
Dim vd() As Variant = v

// returns array of variants with strings
Dim vv As Variant = w.EvaluateJavaScript("o = [ ""a"",""b"",""d"" ] ;", e)
Dim vs() As Variant = vv

// returns dictionary
Dim v3 As Variant = w.EvaluateJavaScript("o = { a:1 } ", e)
Dim dd As Dictionary = v3

Break
```

Notes: Synchronous version which waits for JavaScript to return the value.
Error is set in case of errors.

Result is converted from JavaScript data types to Xojo datatypes, usually variants, dictionaries or array of variants.

18.17.11 goBack

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to the back item in the back-forward list.

18.17.12 goForward

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to the forward item in the back-forward list.

18.17.13 goToBackForwardListItem(Item as WKBackForwardListItemMBS)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to an item from the back-forward list and sets it as the current item.
Notes: item: The item to which to navigate. Must be one of the items in the web view's back-forward list.

18.17.14 handlesURLScheme(urlScheme as String) as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Checks whether or not WKWebViews handle the given URL scheme by default.
Notes: urlScheme: The URL scheme to check.

Returns true if supported.

18.17.15 HTMLText as String

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries HTML text for the current main page.
Notes: This is live, so any changes made after loading will be included.

18.17.16 LoadData(Data as MemoryBlock, MIMEType as String, textEncodingName as String, baseURL as string = "")

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.LoadData "<p>Hello World</p>", "text/html", "UTF-8", ""
```

Notes: data: The data to use as the contents of the webpage.
MIMEType: The MIME type of the data.
characterEncodingName: The data's character encoding name.
baseURL: A URL used to resolve relative URLs within the document.

18.17.17 LoadFileURL(File as FolderItem, readAccessItem as FolderItem)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to a requested file.
Example:

```
Dim w as WKWebViewMBS // your web viewer
Dim HTMLFile as folderItem = getfolderItem("test.html")
```

```
// pass folderitem parent to allow read to other files in same folder
w.LoadFileURL HTMLFile, HTMLFile.parent
```

Notes: Navigates to the requested file URL on the filesystem.
file: The file URL to which to navigate.
readAccessItem: The file or folder to allow read access to.
If readAccessItem references a single file, only that file may be loaded by WebKit.
If readAccessItem references a directory, files inside that file may be loaded by WebKit.

18.17.18 LoadHTML(htmlText as String, baseURL as string = "")

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the webpage contents and base URL.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.LoadHTML "<p>Hello World</p>"
```

Notes: htmlText: The string to use as the contents of the webpage.
baseURL: A URL used to resolve relative URLs within the document.

18.17.19 LoadURL(URL as string)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to a requested URL.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.LoadURL "https://www.mbsplugins.de/xojo"
```

18.17.20 LoadURLRequest(Request as NSURLRequestMBS)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Navigates to a requested URL.

18.17.21 MainResourceData(byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries the source data from the webviewer.
Notes: Returns nil in case of error.
Also sets error parameter to NSErrorMBS for details on the error.

18.17.22 paste

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Pastes clipboard into current position in document.

18.17.23 PDFData(byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Create a PDF document representation from the web page currently displayed in the WKWebView.
Notes: Returns PDF data of new PDF Document.
If the data is written to a file the resulting file is a valid PDF document.

Available in macOS 11.0 or newer.
May also work if Safari 14 is installed on older macOS versions.

18.17.24 PlainText as String

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries plain text for the current main page.
Notes: This is live, so any changes made after loading will be included.

18.17.25 printOperation(printInfo as NSPrintInfoMBS) as NSPrintOperationMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Creates a print operation for the current web viewer.
Example:

```
dim browser as WKWebViewMBS // your web viewer
Dim pi As NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
Dim po As NSPrintOperationMBS = browser.printOperation(pi)

po.showsPrintPanel = True
po.showsProgressPanel = True

po.runOperationModalForWindow(Self)
```

Notes: Works for WebKit 1.x and 2.x.

For WebKit 2.x may run into endless loop for some websites due to bugs in Apple's WebKit framework.

18.17.26 reload

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Reloads the current page.
Example:

```
Public Sub Reload(browser as WKWebViewMBS, force as Boolean)
If force Then
browser.reloadFromOrigin
Else
browser.reload
End If
End Sub
```

18.17.27 reloadFromOrigin

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Reloads the current page, performing end-to-end revalidation using cache-validating conditionals if possible.
Example:

```
Public Sub Reload(browser as WKWebViewMBS, force as Boolean)
If force Then
browser.reloadFromOrigin
Else
browser.reload
End If
```

```
End Sub
```

18.17.28 removeAllUserScripts

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Removes all associated user scripts.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.removeAllUserScripts
```

18.17.29 selectAll

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Selects all.

18.17.30 setMagnification(magnification as double, pointX as double, pointY as double)

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: Scales the page content by a specified factor and centers the result on a specified point.
Example:

```
Dim browser As WKWebViewMBS // your instance

// zoom to 200% on 100/100
browser.setMagnification(2, 100, 100)
```

Notes: magnification: The factor by which to scale the content.
x/y: The point (in view space) to center magnification on. As usual with Cocoa, bottom is y = 0.

For MacOS 64-bit only.

18.17.31 stopLoading

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Stops loading all resources on the current page.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.stopLoading
```

18.17.32 takeSnapshot(byref error as NSErrorMBS) as NSImageMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Requests a snapshot.
Example:

```
// get a webview reference
Dim web As WKWebViewMBS = HTMLViewer1.WKWebViewMBS

// take snapshot
Dim error As NSErrorMBS
Dim image As NSImageMBS = web.takeSnapshot(error)

If error <>Nil Then
MsgBox error.LocalizedDescription
return
End If

// save as PNG
Dim png As String = image.PNGRepresentation

// write to file
Dim f As FolderItem = SpecialFolder.Desktop.Child("test.png")
Dim b As BinaryStream = BinaryStream.Create(f, True)
b.Write png
```

Notes: This is synchronous version which waits for image to be finished.
In case of error the result is nil and error may be set.

18.17.33 userScripts as WKUserScriptMBS()

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The user scripts associated with the user content controller.

18.17.34 WebArchiveData(byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries web archive data for the website.
Example:

```
Dim w As WKWebViewMBS = HTMLViewer1.WKWebViewMBS

Dim error As NSErrorMBS
Dim data As MemoryBlock = w.WebArchiveData(error)

Break // inspect in debugger
```

Notes: May return nil in case of error and maybe report the NSError in case of an error.

18.17.35 Properties

18.17.36 allowFileAccessFromFileURLs as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Allow file access for file URLs.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowFileAccessFromFileURLs = True
```

Notes: Access to files is allowed for some files.
Please check WebKit documentation for details.

Set allowFileAccessFromFileURLs and allowUniversalAccessFromFileURLs to true to disable all the security checks to block local file access for websites. allowUniversalAccessFromFileURLs would allow all file URLs. allowFileAccessFromFileURLs would only allow in same path.

(Read and Write property)

18.17.37 allowsBackForwardNavigationGestures as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether horizontal swipe gestures will trigger back-forward list navigations.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsBackForwardNavigationGestures = True
```

Notes: The default value is false.
(Read and Write property)

18.17.38 allowsLinkPreview as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether pressing on a link displays a preview of the destination for the link.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsLinkPreview = True
```

Notes: (Read and Write property)

18.17.39 allowsMagnification as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether magnify gestures will change the web view,Äôs magnification.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.allowsMagnification = True
```

Notes: The default value is false. You can set the magnification property even if allowsMagnification is set to false.
For MacOS 64-bit only.
(Read and Write property)

18.17.40 allowUniversalAccessFromFileURLs as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Allow universal file access for URLs.
Notes: Any file URL will be loaded if true.
Please check WebKit documentation for details.

Does not work in MacOS 10.14.
(Read and Write property)

18.17.41 backForwardList as WKWebViewMBS

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The web view's back-forward list.
Notes: (Read only property)

18.17.42 backgroundColor as NSColorMBS

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: The background color.
Example:

```
Dim w As WKWebViewMBS = HTMLViewer1.WKWebViewMBS // your webviewer

w.drawsBackground = False
w.backgroundColor = NSColorMBS.clearColor
```

Notes: If background is not drawn and/or transparent, you may see through.
Requires also CSS to have web view page transparent.
(Read and Write property)

18.17.43 CanGoBack as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether there is a back item in the back-forward list that can be navigated to.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "CanGoBack: "+If(Browser.CanGoBack, "yes", "no")
```

Notes: (Read only property)

18.17.44 CanGoForward as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether there is a forward item in the back-forward list that can be navigated to.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "CanGoForward: "+If(Browser.CanGoForward, "yes", "no")
```

Notes: (Read only property)

18.17.45 Configuration as WKWebViewConfigurationMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the configuration used to initialize the WKWebView.
Notes: (Read only property)

18.17.46 customUserAgent as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The custom user agent string.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.customUserAgent = "MyBrowser/1.0"
```

Notes: If no custom user agent string has been set, this is set to "".
Available in 32-bit. Available in 64-bit for macOS 10.11 or newer.
(Read and Write property)

18.17.47 developerExtrasEnabled as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether developer extras are enabled.
Example:

```
Dim browser As WKWebViewMBS // your instance

// enable context menu entry to show developer extras
browser.developerExtrasEnabled = True
```

Notes: Enables the context menu to inspect items.
(Read and Write property)

18.17.48 drawsBackground as Boolean

Plugin Version: 20.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to draw background for the view.
Example:

```
Dim w As WKWebViewMBS = HTMLViewer1.WKWebViewMBS // your webviewer

w.drawsBackground = False
w.backgroundColor = NSColorMBS.clearColor
```

Notes: If background is not drawn and/or transparent, you may see through.
Requires also CSS to have web view page transparent.
(Read and Write property)

18.17.49 EstimatedProgress as Double

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: An estimate of what fraction of the current navigation has been loaded.
Notes: This value ranges from 0.0 to 1.0 based on the total number of bytes expected to be received, including the main document and all of its potential subresources. After a navigation loading completes, the estimatedProgress remains at 1.0 until a new navigation starts, at which point the estimatedProgress is reset to 0.0.
(Read only property)

18.17.50 hasOnlySecureContent as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether all resources on the page have been loaded through securely encrypted connections.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "hasOnlySecureContent: "+If(Browser.hasOnlySecureContent, "yes", "no")
```

Notes: Only set for 64-bit, always false on 32-bit.
(Read only property)

18.17.51 HTTPCookieStore as WKHTTPCookieStoreMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Convenience property to directly query HTTP cookie store for WKWebview configuration.
Example:

```
Dim CookieStore As WKHTTPCookieStoreMBS = WKWebViewControlMBS1.WKWebView.HTTPCookieStore
Dim cookies() As NSHTTPCookieMBS = CookieStore.AllCookies
Dim cookie As NSHTTPCookieMBS = cookies(0)

// now check in debugger
Break
```

Notes: (Read only property)

18.17.52 IsLoading as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value indicating whether the view is currently loading content.
Example:

```
Dim browser As WKWebViewMBS // your instance

// load a page
browser.LoadURL "https://www.mbsplugins.de/xojo"

// wait synchronously for website loading
While browser.IsLoading
DelayMBS 0.1
Wend

MsgBox "loaded"
```

Notes: (Read only property)

18.17.53 javaEnabled as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether java is enabled.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.17.54 javaScriptCanOpenWindowsAutomatically as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether javascript can open new windows.
Example:

```
Dim browser As WKWebViewMBS // your instance

// disable popup windows
Browser.javaScriptCanOpenWindowsAutomatically = False
```

Notes: (Read and Write property)

18.17.55 javaScriptEnabled as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: JavaScript was evaluated.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "JavaScript Enabled: "+If(Browser.javaScriptEnabled, "yes", "no")
```

Notes: Error is only set with 64-bit, not with 32-bit.
Tag is passed from EvaluateJavaScript call.
(Read and Write property)

18.17.56 loadsImagesAutomatically as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether to load images automatically.
Example:

```
Dim browser As WKWebViewMBS // your instance

// disable loading of images
browser.loadsImagesAutomatically = False
```

Notes: (Read and Write property)

18.17.57 magnification as Double

Plugin Version: 20.0, Platform: macOS, Targets: Desktop only.

Function: The factor by which the page content is currently scaled.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.magnification = 2.0
```

Notes: The default value is 1.0.
For MacOS 64-bit only.
(Read and Write property)

18.17.58 mediaType as String

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: The media type for the WKWebView.
Notes: The value of mediaType will override the normal value of the CSS media property.
Setting the value to "" will restore the normal value.
The default value is "".

Available in macOS 11.0 or newer.
(Read and Write property)

18.17.59 MIMEType as String

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Queries current mime type for showing document.
Notes: Should be "text/html" for a website or "application/pdf" for a PDF document.
(Read only property)

18.17.60 minimumFontSize as Double

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum font size to use.
Example:

```
Dim browser As WKWebViewMBS // your instance

browser.minimumFontSize = 20
```

Notes: (Read and Write property)

18.17.61 pageZoom as Double

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: The factor by which page content is scaled relative to the viewport.
Notes: The default value is 1.0.
Changing this value is equivalent to web content setting the CSS "zoom" property on all page content.

Available in macOS 11.0 or newer.
(Read and Write property)

18.17.62 plugInsEnabled as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether plugins are enabled.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "plugInsEnabled: "+If(Browser.plugInsEnabled, "yes", "no")
```

Notes: (Read and Write property)

18.17.63 Preferences as WKPreferencesMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop & iOS.

Function: Convenience property to directly query HTTP cookie store for WKWebview preferences.
Notes: (Read only property)

18.17.64 privateBrowsing as Boolean

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: Whether private browsing is enabled.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox "privateBrowsing: "+If(Browser.privateBrowsing, "yes", "no")
```

Notes: If enabled, no data should be stored for cookies or cache on disk, so it's cleared when web viewer is destroyed.
Default is non-private mode for the web viewer.

It looks like you can't change mode after web viewer was created.
(Read and Write property)

18.17.65 Title as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The page title.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox Browser.Title
```

Notes: (Read only property)

18.17.66 URL as String

Plugin Version: 20.0, Platform: macOS, Targets: Desktop & iOS.

Function: The active URL.
Example:

```
Dim browser As WKWebViewMBS // your instance

MsgBox Browser.URL
```

Notes: This is the URL that should be reflected in the user interface.
(Read only property)

18.18 module WVWebViewModuleMBS

18.18.1 module WVWebViewModuleMBS

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: The module for global options to WKWebView.
Example:

WVWebViewModuleMBS.UsePrivateBrowsing = True

Blog Entries

- MBS Xojo Plugins, version 19.5pr5

18.18.2 Properties

18.18.3 UsePrivateBrowsing as Boolean

Plugin Version: 19.5, Platform: macOS, Targets: Desktop only.

Function: Whether to create WKWebViewControlMBS with private browsing enabled.
Example:

WVWebViewModuleMBS.UsePrivateBrowsing = True

Notes: Set this property to true before creating the WKWebViewControlMBS control, so the plugin can enable private browsing.
Set to false later to create a normal control again.

You may want to set this to true or false before the window opens and control initialized.
(Read and Write property)

Chapter 19

List of Questions in the FAQ

Chapter 20

The FAQ

20.0.1 Can anyone help me convert seconds to time in this format hh:mm:ss?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sure, here's a routine I use (which has an advantage over the previously-posted Date-based solution in that you don't have to rely on the creation of an object – all that happens is some division and string concatenation):
Example:

```
Function SecsToTimeString(timeInSecs as Integer, padHours as boolean, padMinutes as boolean) as string
// Given an amount time (in seconds), generates a string representing that amount
// of time. The padHours and padMinutes parameters determine whether to display
// hours and minutes if their values are zero.

// Examples:
// timeInSecs = 90, padHours = true; returns "00:01:30"
// timeInSecs = 1, padHours = false, padMinutes = true; returns "00:01"
// timeInSecs = 3601, padMinutes = false; returns "01:00:01"

dim hours, minutes, seconds as Integer
dim hoursString, minutesString as string

hours = timeInSecs / 3600
minutes = (timeInSecs mod 3600) / 60
seconds = timeInSecs mod 60

if hours = 0 then
if padHours then
hoursString = "00:"
else
hoursString = ""
end if
```

```
else
hoursString = Format(hours, "##\:")
end if
if minutes = 0 then
if hours <>0 or padMinutes then
minutesString = "00:"
else
minutesString = ""
end if
else
minutesString = Format(minutes, "00\:")
end if

return hoursString + minutesString + Format(seconds, "00")
End Function
```

Notes: (from the rb mailinglist)

20.0.2 Do you have plugins for Android?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Since there is no plugin SDK for Android, we have no way to make a plugin for Android.
Notes: We support macOS, Windows, Linux and iOS.

20.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use functions from NSColor to get proper highlight color in RGB:
Example:

```
Function ProperHighlightColor(active as Boolean) As Color
#if TargetCocoa
Dim theColor As NSColorMBS
If active Then
theColor = NSColorMBS.alternateSelectedControlColor
Else
theColor = NSColorMBS.secondarySelectedControlColor
End If

Dim rgbColor As NSColorMBS = theColor.colorUsingColorSpaceName(NSColorSpaceMBS.NSCalibrate-
```

```
dRGBColorSpace)
If rgbColor <>Nil Then
Dim red as Integer = rgbColor.redComponent * 255.0
Dim green as Integer = rgbColor.greenComponent * 255.0
Dim blue as Integer = rgbColor.blueComponent * 255.0
Return RGB(red, green, blue)
Else
Return HighlightColor
End If
#else
return HighlightColor
#endif
End Function
```

Notes: As you see we convert color to Calibrated RGB for best results.
See also:

20.0.4 How to catch delete key?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code in keydown event catches delete or backspace keys.
Example:

```
Function KeyDown(Key As String) As Boolean
if asc(key) = 8 or asc(key) = 127 then
MsgBox "Delete"
Return true
end if
End Function
```

See also:

20.0.5 How to convert cmyk to rgb?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

The following is the code to convert cmyk values to an RGB color datatype.
It's just a basic estimate of the color values. If you are looking for completely color accurate solution, this is not it. It should work for most people. :)
Example:

```
Function CMYKToRGB(c as Integer, m as Integer, y as Integer, k as Integer) As color
// converts c,m,y,k values (0-100) to color data type RGB
// place this in a method. Supply C,M,Y,K values-
// it returns color datatype


dim color_RGB as color
dim r, g, b as Integer

r=255-round(2.55*(c+k))
if r<0 then
r=0
end if
g=255-round(2.55*(m+k))
if g<0 then
g=0
end if
b=255-round(2.55*(y+k))
if b<0 then
b=0
end if

color_RGB=RGB(r,g,b)

return color_RGB

End Function
```

Notes:

(from the rb mailinglist)
See also:

20.0.6 How to delete a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code deletes a folder recursively.
Example:

```
Sub deletefolder(f as folderitem)
dim files(-1) as FolderItem

if f=nil then Return

// delete single file
if f.Directory=false then
f.Delete
Return
end if

// get a list of all items in that folder
dim i,c as Integer
c=F.Count
for i=1 to c
files.Append f.TrueItem(i)
next

// delete each item
for each fo as FolderItem in files
if fo=nil then
' ignore
elseif fo.Directory then
deletefolder fo
fo.delete
else ' file
```

```
fo.Delete
end if
next

f.Delete
End Sub
```

See also:

- 20.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1466
- 20.0.4 How to catch delete key? 1467
- 20.0.5 How to convert cmyk to rgb? 1468
- 20.0.7 How to detect if CPU if 64bit processor? 1470
- 20.0.8 How to query variant type string for a variant? 1471
- 20.0.9 How to refresh a htmlviewer on Windows? 1472

20.0.7 How to detect if CPU if 64bit processor?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Via CPUID you can ask CPU:
Example:

```
dim c as new CPUIDMBS

if c.Flags(CPUIDMBS.kFeatureLM) then
MsgBox "64-bit CPU"
else
MsgBox "32-bit CPU"
end if
```

Notes: Should work on all intel compatible CPUs.
See also:

- 20.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1466
- 20.0.4 How to catch delete key? 1467
- 20.0.5 How to convert cmyk to rgb? 1468
- 20.0.6 How to delete a folder? 1469
- 20.0.8 How to query variant type string for a variant? 1471
- 20.0.9 How to refresh a htmlviewer on Windows? 1472

20.0.8 How to query variant type string for a variant?

Plugin Version: 20.5, Platforms: macOS, Linux, Windows.

Answer: The following example function returns type string for variant.
Example:

```
Public Function VariantTypeString(v as Variant) as string
// Xojo's VarType doesn't know Unsigned integers
'Dim type As Integer = VarType(v)

// MBS VarType can detect unsigned integer
Dim type As Integer = GetVariantTypeMBS(v)

Dim IsArray As Boolean = BitwiseAnd(type, Variant.TypeArray) = Variant.TypeArray

// type without array
type = BitwiseAnd(type, Bitwise.OnesComplement(Variant.TypeArray))

// build a dictionary to map types on first call
Static TypeMap As Dictionary
If TypeMap = Nil Then
TypeMap = New Dictionary
TypeMap.Value(Variant.TypeBoolean) = "Boolean"
TypeMap.Value(Variant.TypeCFStringRef) = "CFStringRef"
TypeMap.Value(Variant.TypeColor) = "Color"
TypeMap.Value(Variant.TypeCString) = "CString"
TypeMap.Value(Variant.TypeCurrency) = "Currency"
TypeMap.Value(Variant.TypeDate) = "Date"
TypeMap.Value(Variant.TypeDateTime) = "DateTime"
TypeMap.Value(Variant.TypeDouble) = "Double"
TypeMap.Value(Variant.TypeInt32) = "Int32"
TypeMap.Value(Variant.TypeInt64) = "Int64"
TypeMap.Value(Variant.TypeInteger) = "Integer"
TypeMap.Value(Variant.TypeNil) = "Nil"
TypeMap.Value(Variant.TypeObject) = "Object"
TypeMap.Value(Variant.TypeOSType) = "OSType"
TypeMap.Value(Variant.TypePString) = "PString"
TypeMap.Value(Variant.TypePtr) = "Ptr"
TypeMap.Value(Variant.TypeSingle) = "Single"
TypeMap.Value(Variant.TypeString) = "String"
TypeMap.Value(Variant.TypeStructure) = "Structure"
TypeMap.Value(Variant.TypeText) = "Text"
TypeMap.Value(Variant.TypeWindowPtr) = "WindowPtr"
TypeMap.Value(Variant.TypeWString) = "WString"

// MBS extra types
TypeMap.Value(Variant.TypeInt32+100) = "UInt32"
TypeMap.Value(Variant.TypeInt64+100) = "UInt64"
```

```
End If

// lookup type

#if DebugBuild then
If Not TypeMap.HasKey(type) Then
Break // missing type
End If
#endif

If IsArray Then
Return "Array of " + TypeMap.Lookup(type,"?")
Else
Return TypeMap.Lookup(type,"?")
End If
End Function
```

See also:

- 20.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1466
- 20.0.4 How to catch delete key? 1467
- 20.0.5 How to convert cmyk to rgb? 1468
- 20.0.6 How to delete a folder? 1469
- 20.0.7 How to detect if CPU if 64bit processor? 1470
- 20.0.9 How to refresh a htmlviewer on Windows? 1472

20.0.9 How to refresh a htmlviewer on Windows?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can ask the browser to reload the website with this code line:
Example:

```
call htmlViewer1.IERunJavaScriptMBS("javascript:document.location.reload()")
```

See also:

- 20.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1466
- 20.0.4 How to catch delete key? 1467
- 20.0.5 How to convert cmyk to rgb? 1468

20.0.10 Is there an example for vector graphics in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this example inside the paint event of a window:
Example:

```
dim v as Group2D
dim r as RectShape
dim s as StringShape

const pi=3.14

s=new StringShape
s.Text="Hello World!"
s.TextFont="Geneva"
s.TextSize=24
s.FillColor=rgb(0,0,255)
s.Italic=true
s.y=5
s.x=0

r=new RectShape

r.X=0
r.y=0
r.Height=100
r.Width=180
r.BorderColor=rgb(255,0,0)
r.FillColor=rgb(0,255,0)
r.BorderWidth=5
r.Border=50

v=new Group2d
v.Append r
v.Append s
v.Rotation=pi*-20.0/180.0
v.x=150
v.y=150

g.DrawObject v
```

20.0.11 Picture functions do not preserve resolution values?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, the picture functions return pictures with no/default resolution values.
Example:

```
dim l as Picture = LogoMBS(500)

l.HorizontalResolution = 300
l.VerticalResolution = 300

dim r as Picture = l.Rotate90MBS

MsgBox str(r.HorizontalResolution)+" x "+str(r.VerticalResolution)

r.HorizontalResolution = l.HorizontalResolution
r.VerticalResolution = l.VerticalResolution

MsgBox str(r.HorizontalResolution)+" x "+str(r.VerticalResolution)
```

Notes: So please fix them yourself after calling a function.

Maybe in the future this changes, but currently you can't really set this easily from plugin code.

20.0.12 A toolbox call needs a rect - how do I give it one?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Fill a memoryblock like this:
Example:

```
Dim MB As Memoryblock
MB = NewMemoryBlock(8)
MB.Short(0) = window1.Top
MB.Short(2) = window1.Left
MB.Short(4) = window1.Height+window1.Top // bottom
MB.Short(6) = window1.Width+window1.Left // right
```

20.0.13 API client not supported?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you get this exception message on SQLConnectionMBS.Connect, we may have a problem.
Notes: First case is that the given thing is not supported (e.g. MS SQL directly on Mac).
Second case is that the plugin compilation went wrong and the support for the database was not linked into the plugin. Like MySQL missing or MS SQL on Windows missing. In that case please contact us to fix the plugin.

20.0.14 Can I access Access Database with Java classes?

Plugin Version: all, Platform: Windows.

Answer: You can use ucanaccess to access databases created with Microsoft
Example:

```
dim options(-1) as string

// load all the jar files we have in a folder called java:

dim appFolder as FolderItem = GetFolderItem("")

Dim count as Integer = appFolder.Parent.Child("java").Count
dim libjs() as string
For i as Integer = 1 to count
Dim f As FolderItem = appFolder.Parent.Child("java").item(i)
If f <>Nil and f.Exists Then
libjs.append f.NativePath+";"
End If
Next

// now init virtual machine
dim librery as string = Join(libjs, "")
dim vm as new JavaVMMBS(librery)

if vm.Handle = 0 then
MsgBox "Failed to initialize virtual machine"
else
// now make a new database connection with ucanaccess
dim d as new JavaDatabaseMBS(vm,"net.ucanaccess.jdbc.UcanaccessDriver")
Dim DbFile as FolderItem = appFolder.Parent.Child("Database11.accdb")
dim j as JavaConnectionMBS = d.getConnection("jdbc:ucanaccess://"+DbFile.NativePath)

// select and show values
dim r as JavaResultSetMBS = j.MySelectSQL("Select * From test")
while r.NextRecord
MsgBox r.getString("FirstName") +" "+ r.getString("LastName")
wend

end if
```

```
Exception e as JavaExceptionMBS
MsgBox e.message+" errorcode: "+str(e.ErrorNumber)
```

Notes: see website:
http://ucanaccess.sourceforge.net/site.html

20.0.15 Can I create PDF from Xojo Report using DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, we have a graphics class integration for DynaPDF.
Notes: Since MBS Plugin in version 19.2, we can integrate reports with Xojo.

20.0.16 Can I use AppleScripts in a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, but they run on the server, not on the client.
Example:

```
dim a as new AppleScriptMBS

// query my application name
a.Compile "tell application ""System Events"" to return name of current application"

// run
a.Execute

// show result
label1.text = a.Result

// shows something like "My Application.fcgi.debug"
```

Notes: This can be useful to control the server from remote, if and only if the your sever is running Mac OS X.

20.0.17 Can I use graphics class with DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sorry, no. We can't provide a graphics subclass from plugin.
Notes: The is a feature request to allow graphics subclasses:
Feedback case 11391: feedback://showreport?report_id=11391

20.0.18 Can I use sockets on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, but they run on the server, not on the client.
Notes: You can use HTTPSocket, SMTPSocket, POP3Socket, SMTPSecureSocket, SecurePOP3Socket, EasyTCPSocket, EasyUDPSocket, AutoDiscovery, our Bonjour classes or our CURL* classes. But all of them work on the server, not on the client.

This means if you search for a printer with Bonjour, you can find the printers in the local network on your server hosting site. Using SMTPSocket may be a good idea for sending emails from the server like notifications.

20.0.19 Can I use your ChartDirector plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our ChartDirector plugin works just fine on the Xojo Web Edition.
Example:

```
// The data for the pie chart
dim data(-1) as Double=array(55.0, 18.0, 25.0, 22.0, 18.0, 30.0, 35.0)

// The labels for the pie chart, Words are choosen random to check font!
dim labels(-1) as string=array("Germany","Italy","France","Spain","UK","Poland","Russia")

// The colors to use for the sectors
dim colors(-1) as Integer

colors.Append &h66aaee
colors.Append &heebb22
colors.Append &hbbbbbb
colors.Append &h8844ff

if TargetLinux then
CDBaseChartMBS.SetFontSearchPath "/usr/share/fonts/truetype/msttcorefonts"
end if

// Create a PieChart object of size 360 x 300 pixels
dim c as new CDPieChartMBS(700, 600)
```

```
c.setBackground(c.linearGradientColor(0, 0, 0, c.getHeight(), &h0000cc, &h000044))
c.setRoundedFrame(&hffffff, 16)
dim tt as CDTextBoxMBS = c.addTitle("ChartDirector Demonstration", "timesbi.ttf", 18)
tt.setMargin(0, 0, 16, 0)
tt.setFontColor(&hFFFFFF)

// Set the center of the pie at (180, 140) and the radius to 100 pixels
c.setPieSize 350,300,150
// Set the sector colors
c.setColors(c.kDataColor, colors)

// Draw the pie in 3D with a pie thickness of 20 pixels
c.set3D(20)

dim t as CDTextBoxMBS = c.setLabelStyle("arialbd.ttf", 10, &h000000)
t.setBackground(CDPieChartMBS.kSameAsMainColor, CDPieChartMBS.kTransparent, CDPieChartMBS.soft-
Lighting(CDPieChartMBS.kRight, 0))
t.setRoundedCorners(8)

// Use local gradient shading for the sectors, with 5 pixels wide
// semi-transparent white (bbffffff) borders
c.setSectorStyle(CDPieChartMBS.kLocalGradientShading, &hbbffffff, 0)

// Set the pie data and the pie labels
c.setData data,labels
call c.setLabelStyle "arialbd.ttf",18

dim pic as picture = c.makeChartPicture
dim wp as new WebPicture(pic, Picture.FormatJPEG) // JPEG makes it smaller and faster

ImageView1.Picture=wp
```

Notes: Be aware that our plugin produces pictures for you, which you assign to ImageViews. Tranfserring those pictures takes time, so you can optimize that with using WebPicture class. There you can decide between different compressions to improve speed (use JPEG instead of PNG).

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

20.0.20 Can I use your DynaPDF plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our DynaPDF plugin works just fine on the Xojo Web Edition.
Notes: PDF files are created on the server. You may want to offer a preview to the user which uses reduced resolution images to reduce the time to download the PDF.

See our Create PDF example for the Xojo Web Edition.

20.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

20.0.22 Can you get an unique machine ID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: There is nothing like an unique machine ID.
Notes: 1:
You can use the MAC IDs of the network interfaces.
This can be changed by the user with software tools.
And the list of network interfaces changes if user reorder the interfaces.

2:
You can use the system folder creation date/time.
This may stay equal after cloning machines or after migration to new PC.

3:
You can use the Mac Serialnumber.
Mac only and it can happen that a Mac does not have a serial number.

4:
You can use the x86 CPU ID.
This is x86 CPU only and does not avoid running on the same CPU in different PCs.

20.0.23 ChartDirector: Alignment Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Alignment Specification
Notes: In many ChartDirector objects, you may specify the alignment of the object's content relative to its boundary. For example, for a TextBox object, you may specify the text's alignment relative to the box boundary by using TextBox.setAlignment.

The ChartDirector API defines several constants for the alignment options.

ConstantValueDescription

BottomLeft	1	The leftmost point on the bottom line.
BottomCenter	2	The center point on the bottom line.
BottomRight	3	The rightmost point on the bottom line.
Left	4	The leftmost point on the middle horizontal line.
Center	5	The center point on the middle horizontal line.
Right	6	The rightmost point on the middle horizontal line.
TopLeft	7	The leftmost point on the top line.
TopCenter	8	The center point on the top line.
TopRight	9	The rightmost point on the top line.
Bottom	2	The center point on the bottom line. Same as BottomCenter.
Top	8	The center point on the top line. Same as TopCenter.
TopLeft2	10	An alternative top-left position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, TopLeft2 refers to refers to the left of the top side, while TopLeft refers to the top of the left side. The reverse applies for a horizontal axis.
TopRight2	11	An alternative top-right position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, TopRight2 refers to refers to the right of the top side, while TopRight refers to the top of the right side. The reverse applies for a horizontal axis.
BottomLeft2	12	An alternative bottom-left position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, BottomLeft2 refers to refers to the left of the bottom side, while BottomLeft refers to the bottom of the left side. The reverse applies for a horizontal axis.
BottomRight2	13	An alternative bottom-right position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, BottomRight2 refers to refers to the right of the bottom side, while BottomRight refers to the bottom of the right side. The reverse applies for a horizontal axis.

20.0.24 ChartDirector: Color Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Color Specification
Notes: Many functions in the ChartDirector API accept colors as parameters. ChartDirector supports col-

ors specified in web and HTML compatible ARGB format, in which ARGB refers to the Alpha transparency, Red, Green and Blue components of the color.

In addition to ARGB colors, ChartDirector supports ”dynamic” colors. A dynamic color is a color that changes depending on the position of the pixels. The ”dynamic” colors that ChartDirector supports include ”pattern colors”, ”metal colors”, ”gradient colors”, ”zone colors” and ”dash line colors”.

ChartDirector supports specifying colors indirectly using ”palette colors”. When a ”palette color” is used, the color is specified as an index to a palette. The actual color is looked up from the palette.ARGB Color
ARGB color consists of 4 components - alpha transparency, red, green and blue. The four components are encoded as a 32-bit number, with each component occupying 8 bits. In hexadecimal notation, it is AARRGGBB, where AA, RR, GG and BB are the alpha transparency, red, green and blue components.

Each component ranges from 00 - FF (0 - 255), representing its intensity. For example, pure red color is 00FF0000, pure green color is 0000FF00, and pure blue color is 000000FF. White color is 00FFFFFF, and black color is 00000000.

Most programming language requires you to put special prefix in front of hexadecimal characters. For C++, the prefix is ”0x”. For example, the syntax for the hexadecimal number 00FFFFFF is 0x00FFFFFF, or simply 0xFFFFFF.
For the alpha transparency component, a zero value means the color is not transparent all at. This is equivalent to traditional RGB colors. A non-zero alpha transparency means the the color is partially transparent. The larger the alpha transparency, the more transparent the color will be. If a partially transparent color is used to draw something, the underlying background can still be seen.

For example, 80FF0000 is a partially transparent red color, while 00FF0000 is a non-transparent red color.

Note that ChartDirector’s ARGB color is web and HTML compatible. For example, red is FF0000, the same as in HTML. There are many resources on the web that provide tables in which you can click a color and it will show its HTML color code. These color codes can be used in ChartDirector.

If alpha transparency is FF (255), the color is totally transparent. That means the color is invisible. It does not matter what the RGB components are. So in ChartDirector, only one totally transparent color is used - FF000000. All other colors of the form FFnnnnnn are reserved to represent palette colors and dynamic colors, and should not be interpreted as the normal ARGB colors.

The totally transparent color FF000000 is often used in ChartDirector to disable drawing something. For example, if you want to disable drawing the border of a rectangle, you can set the border color to totally transparent.

For convenience, ChartDirector defines a constant called Transparent, which is equivalent to FF000000.Pattern Color

A pattern color is a dynamic color that changes according to a 2D periodic pattern. When it is used to fill an area, the area will look like being tiled with a wallpaper pattern.

Pattern colors are created using BaseChart.patternColor, BaseChart.patternColor2, DrawArea.patternColor and DrawArea.patternColor2. The patternColor method creates pattern colors using an array of colors as a bitmap. The patternColor2 method creates pattern colors by loading the patterns from image files.

These methods return a 32-bit integer acting as a handle to the pattern color. The handle can be used in any ChartDirector API that expects a color as its input.Metal Color
A metal color is a color of which the brightness varies smoothly across the chart surface as to make the surface looks shiny and metallic. ChartDirector supports using any color as the base color of the metal color. In particular, using yellow and grey as the base colors will result in metal colors that look gold and silver.

Metal colors are most often used as background colors of charts. They are created using CDBaseChartMBS.metalColor, CDBaseChartMBS.goldColor and CDBaseChartMBS.silverColor. The first method allows you to specify an arbitrary base color. The second and third methods use yellow and grey as the base colors, resulting in gold and silver metal colors.

These methods return a 32-bit integer acting as a handle to the gradient color. The handle can be used in any ChartDirector API that expects a color as its input.Gradient Color
A gradient color is a color that changes progressively across a direction.

Gradient colors are created using BaseChart.gradientColor, BaseChart.gradientColor2, DrawArea.gradientColor and DrawArea.gradientColor2. The gradientColor method creates a 2-point gradient color that changes from color A to color B. The gradientColor2 method creates a multi-point gradient colors that changes from color A to B to C ....

These methods return a 32-bit integer acting as a handle to the gradient color. The handle can be used in any ChartDirector API that expects a color as its input.

One common use of multi-point gradient colors is to define colors that have metallic look and feel. Please refer to DrawArea.gradientColor2 for details.Dash Line Colors
A dash line color is a color that switches on and off periodically. When used to draw a line, the line will appear as a dash line.

Dash line colors are created using BaseChart.dashLineColor and DrawArea.dashLineColor. They accept a line color and a dash pattern code as arguments, and return a 32-bit integer acting as a handle to the dash line color. The handle can be used in any ChartDirector API that expects a color as its input.Zone Colors
A zone color is for XY charts only. It is a color that automatically changes upon reaching a data threshold value along the x-axis or y-axis. Zone colors are created using Layer.xZoneColor, Layer.yZoneColor, XYChart.xZoneColor or XYChart.yZoneColor.Palette Colors
Palette colors are colors of the format FFFFnnnn, where the least significant 16 bits (nnnn) are the index to the palette. A palette is simply an array of colors. For a palette color, the actual color is obtained by

looking up the palette using the index. For example, the color FFFF0001 is the second color in the palette (first color is index 0).

The colors in the palette can be ARGB colors or "dynamic" colors (pattern, gradient and dash line colors).

The first eight palette colors have special significance. The first three palette colors are the background color, default line color, and default text color of the chart. The 4th to 7th palette colors are reserved for future use. The 8th color is a special dynamic color that is equal to the data color of the "current data set".

The 9th color (index = 8) onwards are used for automatic data colors. For example, in a pie chart, if the sector colors are not specified, ChartDirector will automatically use the 9th color for the first sector, the 10th color for the second sector, and so on. Similarly, for a multi-line chart, if the line colors are not specified, ChartDirector will use the 9th color for the first line, the 10th color for the second line, and so on.

The ChartDirector API defines several constants to facilitate using palette colors.

ConstantValueDescription

Palette	FFFF0000	The starting point of the palette. The first palette color is (Palette + 0). The nth palette color is (Palette + n - 1).
BackgroundColor	FFFF0000	The background color.
LineColor	FFFF0001	The default line color.
TextColor	FFFF0002	The default text color.
[Reserved]	FFFF0003 - FFFF0006	These palette positions are reserved. Future versions of ChartDirector may use these palette positions for colors that have special significance.
SameAsMainColor	FFFF0007	A dynamic color that is equal to the data color of the current data set. This color is useful for objects that are associated with data sets. For example, in a pie chart, if the sector label background color is SameAsMainColor, its color will be the same as the corresponding sector color.
DataColor	FFFF0008	The starting point for the automatic data color allocation.

When a chart is created, it has a default palette. You may modify the palette using BaseChart.setColor, BaseChart.setColors, or BaseChart.setColors2.

The advantages of using palette colors are that you can change the color schemes of the chart in one place. ChartDirector comes with several built-in palettes represented by the following predefined constants.

ConstantDescription

20.0.25 ChartDirector: Font Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

defaultPalette	An array of colors representing the default palette. This palette is designed for drawing charts on white backgrounds (or lightly colored backgrounds).
whiteOnBlackPalette	An array of colors useful for drawing charts on black backgrounds (or darkly colored backgrounds).
transparentPalette	An array of colors useful drawing charts on white backgrounds (or lightly colored backgrounds). The data colors in this palette are all semi-transparent.

Answer: ChartDirector: Font Specification
Notes: Font Name
In ChartDirector, the font name is simply the file name that contains the font. For example, under the Windows platform, the "Arial" font is "arial.ttf", while the "Arial Bold" font is "arialbd.ttf".

NOTE: Mac OS X Specific Information
In Mac OS X, in addition to ".ttf", ChartDirector also supports Mac OS X font file formats, such as Font Suitcase files and Datafork files (.dfont). These files often contain multiple fonts. For example, the "GillSans.dfont" file contains 6 fonts.

So in addition to the file name, an index is needed to determine the font. The index is specified by appending a " | " character to the font name, followed by the index number. For example, the third font in "GillSans.dfont" is denoted as "GillSans.dfont | 2". (Note: The first font starts at 0.) If no index number is provided, the first font is assumed.

ChartDirector also supports using Mac OS X Font Manager names. For example, one may use "Gill Sans Light Italic" instead of using "GillSans.dfont | 1" as the font name. However, the Mac OS X Font Manager is active only if someone has logged into the Mac GUI console, so this method is only recommended for developing applications that run on the GUI console.

The sample programs that come with ChartDirector are designed to run on all operating systems, so they use generic font file names (eg. "arial.ttf") instead of Mac OS X specific names. To allow them to run on Mac OS X, ChartDirector on Mac OS X has a built-in table to map common font file names to Mac OS X font names:

"arial.ttf", "arialbd.ttf", "ariali.ttf" and "arialbi.ttf" are mapped to "Arial | 0" (Arial), "Arial | 1" (Arial Bold), "Arial | 2" (Arial Italic) and "Arial | 3" (Arial Bold Italic)

"times.ttf", "timesbd.ttf", "timesi.ttf" and "timesbi.ttf" are mapped to "Times New Roman | 0" (Times New Roman), "Times New Roman | 1" (Times New Roman Bold), "Times New Roman | 2" (Times New Roman Italic) and "Times New Roman | 3" (Times New Roman Bold Italic)

"cour.ttf", "courbd.ttf", "couri.ttf" and "courbi.ttf" are mapped to "Courier New | 0" (Courier New), "Courier New | 1" (Courier New Bold), "Courier New | 2" (Courier New Italic) and "Courier New | 3" (Courier New Bold Italic)

Font Location
ChartDirector on Windows does not come with any font files. It relies on the operating system's font files in the " [windows] \Fonts" directory. To see what fonts are installed in your operating system and their file names, use the File Explorer to view that directory.

ChartDirector on Windows will also search for the font files in the "fonts" subdirectory (if it exists) under the directory where the ChartDirector DLL "chartdir.dll" is installed. This is useful for private fonts. Also, for some especially secure web servers, the web anonymous user may not have access to the " [windows] \Fonts" directory. In this case, you may copy the font files to the above subdirectory.

ChartDirector on Mac OS X relies on operating system font files in "/Library/Fonts" and "/System/Library/Fonts".

ChartDirector on Linux, FreeBSD and Solaris assume the fonts files are in the "fonts" subdirectory under the directory where the ChartDirector shared object "libchartdir.so" is installed. ChartDirector on Linux, FreeBSD and Solaris come with a number of font files in the "fonts" subdirectory.

To keep the download size small, ChartDirector on Linux, FreeBSD and Solaris only come with some commonly used fonts. You may download additional fonts from the Internet. In particular, the Microsoft fonts at
http://sourceforge.net/project/showfiles.php?group_id=34153&release_id=105355
is highly recommended. Please refer to
http://www.microsoft.com/typography/faq/faq8.htm
on how you could use the fonts legally in your system.

ChartDirector supports True Type fonts (.ttf), Type 1 fonts (.pfa and .pfb) and Windows bitmap fonts (.fon). On Mac OS X, ChartDirector also supports Font Suitcase and Datafork (.dfont) files. On Linux, FreeBSD and Solaris, ChartDirector also supports Portable Compiled Fonts (.pcf fonts).

If you want ChartDirector to search other directories for the font files, you may list the directories in an environment variable called "FONTPATH".

If you specify an absolute path name for the font file, ChartDirector will use the absolute path name and will not search other directories.Artificial Boldening and Italicizing
Whereas most popular font comes with different styles for "normal", "bold", "italic" and "bold italic", some fonts only come with one style (the normal style). For example, the Monotype Corsiva font that comes with MS Office only has the normal style (mtcorsva.ttf). For these cases, you may append the "Bold" and/or "Italic" words after the font file name (separated with a space) to ask ChartDirector to artificially bolden and/or italicize the font. For example, you may specify the font name as "mtcorsva.ttf Bold".Font List
Instead of specifying a single font file as the font name, you may specify a list of font files as the font name, separated by semi-colons. This is useful when using international characters that are only available in some fonts.

For example, if you would like to use the Arial font ("arial.ttf") for western characters, and the MingLiu font "mingliu.ttc" for Chinese characters (since the Arial font does not have Chinese characters), you may specify the font name as "arial.ttf;mingliu.ttc". In this case, ChartDirector will try the Arial font first. If it cannot find a certain character there, it will try the MingLiu font.Indirect Font Names
ChartDirector supports several special keywords for specifying the font name indirectly. When these keywords are used as font names, ChartDirector will look up the actual font names from a font table. The keywords are as follows:

KeywordsDescription

"normal"	This default normal font, which is the first font in the font table. This is initially mapped to "arial.ttf" (Arial).
"bold"	The default bold font, which is the second font in the font table. This is initially mapped to "arialbd.ttf" (Arial Bold).
"italic"	The default italic font, which is the third font in the font table. This is initially mapped to "ariali.ttf" (Arial Italic).
"boldItalic"	The default bold-italic font, which is the fourth font in the font table. This is initially mapped to "arialbi.ttf" (Arial Bold Italic).
"fontN"	The (N + 1)th font in the font table (the first font is "font0").

The font table can be modified using BaseChart.setFontTable or DrawArea.setFontTable.

The advantage of using indirect font names is that you can change the fonts fonts in your charts in one place.Font Index
Most font files contain one font. However, it is possible a font file contains multiple fonts (that is, a font collection). For example, in True Type fonts, font files with extension ".ttc" may represent a font collection.

If a font file contains multiple font, the font index can be used to specify which font to use. By default, the font index is 0, which means the first font in the font file will be used.Font Size
The font size decides how big a font will appear in the image. The font size is expressed in a font unit called points. This is the same unit used in common word processors.

Instead of specifying font size, some ChartDirector API (eg. TextBox.setFontSize) allow you to specify font height and font width separately. You may use different point sizes for font height and font width to create special effects.Font Color
This is the color to draw the font. (See Color Specification on how colors are represented in ChartDirector.)Font Angle
This is the angle in degrees by which the font should be rotated anti-clockwise.Vertical Layout
By default, text are laid out horizontally, with characters being drawn from left to right.

ChartDirector also supports vertical layout, with characters being drawn from top to bottom. For example, you may use BaseChart.addText to add text that are laid out vertically. Vertical layout is common for

oriental languages such as Chinese, Japanese and Korean.

20.0.26 ChartDirector: Mark Up Language

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Mark Up Language
Notes: ChartDirector Mark Up Language (CDML) is a language for including formatting information in text strings by marking up the text with tags.

CDML allows a single text string to be rendered using multiple fonts, with different colors, and even embed images in the text.Font Styles
You can change the style of the text by using CDML tags. For example, the line:

<*font=timesi.ttf,size=16,color=FF0000>Hello <*font=arial.ttf,size=12,color=8000*>world!
will result in the following text rendered:

In general, all tags in CDML are enclosed by <* and *>. Attributes within the tags determine the styles of the text following the tags within the same block.

If you want to include <* in text without being interpreted as CDML tags, use «* as the escape sequence.

The following table describes the supported font style attributes in CDML. See Font Specification for details on various font attributes.

AttributeDescription

Set the following text to be in superscript style. This attribute does not need to have a value. (You may use ”super” as the attribute instead of ”super=1”.)

Note that unlike HTML tags, no double or single quotes are used in the tags. It is because CDML tags are often embedded as string literals in source code. The double or single quotes, if used, will conflict with the string literal quotes in the source code. Therefore in CDML, no quotes are necessary and they must not be used.

Also, unlike HTML tags, CDML uses the comma character as the delimiter between attributes. It is because certain attributes may contain embed spaces (such as the font file name). So space is not used as the delimiter and the comma character is used instead.

Note the font attribute above starts a new style section, while other attributes just modify the current style

font	Starts a new style section, and sets the font name. You may use this attribute without a value (that is, use "font" instead of "font=arial.ttf") to create a new style section without modifying the font name.
size	The font size.
width	The font width. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
height	The font height. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
color	The text color in hex format.
bgColor	The background color of the text in hex format.
underline	The line width of the line used to underline the following characters. Set to 0 to disable underline.
sub	Set the following text to be in subscript style. This attribute does not need to have a value. (You may use "sub" as the attribute instead of "sub=1".)
super	Set the following text to be in superscript style.

xoffset	Draw the following the text by shifting the text horizontally from the original position by the specified offset in pixels.
yoffset	Draw the following the text by shifting the text vertically from the original position by the specified offset in pixels.
advance	Move the cursor forward (to the right) by the number of pixels as specified by the value this attribute.
advanceTo	Move the cursor forward (to the right) to the position as specified by the value this attribute. The position is specified as the number of pixels to the right of the left border of the block. If the cursor has already passed through the specified position, the cursor is not moved.

section. You may use <*/font*>to terminate a style section, which will restore the font styles to the state before the style section.Blocks and Lines
In CDML, a text string may contain multiple blocks. A block may contain multiple lines of text by separating them with new line characters ("\n") or with <*br*>. The latter is useful for programming languages that cannot represent new line characters easily.

For example, the line:

<*size=15*><*block*><*color=FF*>BLOCK<*br*>ONE<*/*>and <*block*><*color=FF00*>BLOCK<*br*>TWO<
will result in the following text rendered:

The above example contains a line of text. The line contains two blocks with the characters " and " in between. Each block in turn contains two lines. The blocks are defined using <*block*>as the start tag and

<*/*>as the end tag.

When a block ends, font styles will be restored to the state before entering the block.Embedding Images
CDML supports embedding images in text using the following syntax:

<*img=my_image_file.png*>
where my_image_file.png is the path name of the image file.

For example, the line:

<*size=20*>A <*img=sun.png*>day
will result in the following text rendered:

ChartDirector will automatically detect the image file format using the file extension, which must either png, jpg, jpeg, gif, wbmp or wmp (case insensitive).

Please refer to BaseChart.setSearchPath or DrawArea.setSearchPath on the directory that ChartDirector will search for the file.

The <*img*>tag may optionally contain width and height attributes to specify its pixel width and height. In this case, ChartDirector will stretch or compress the image if necessary to the required width and height.Blocks Attributes
CDML supports nesting blocks, that is, a block can contain other sub-blocks. Attributes are supported in the <*block*>tag to control the alignment and orientation of the sub-blocks. The <*img=my_image_file.png*>is treated as a block for layout purposes.

For example, the line:

<*block,valign=absmiddle*><*img=molecule.png*><*block*>Hydrazino\nMolecule<*/*><*/*>
will result in the following text rendered:

The the above starts <*block,valign=absmiddle*>which specifies its content should align with each others in the vertical direction using the absolute middle alignment. The block contains an image, followed by a space characters, and then another block which has two lines of text.

The following table describes the supported attributes inside <*block*>tag:

AttributeDescription

The value baseline means the baseline of sub-blocks should align with the baseline of the block. The baseline

width	The width of the block in pixels. By default, the width is automatically determined to be the width necessary for the contents of the block. If the width attribute is specified, it will be used as the width of the block. If the width is insufficient for the contents, the contents will be wrapped into multiple lines.
height	The height of the block in pixels. By default, the height is automatically determined to be the height necessary for the contents of the block. If the height attribute is specified, it will be used as the height of the block.
maxwidth	The maximum width of the block in pixels. If the content is wider than maximum width, it will be wrapped into multiple lines.
truncate	The maximum number of lines of the block. If the content requires more than the maximum number of lines, it will be truncated. In particular, if truncate is 1, the content will be truncated if it exceeds the maximum width (as specified by maxwidth or width) without wrapping. The last few characters at the truncation point will be replaced with "...".
linespacing	The spacing between lines as a ratio to the default line spacing. For example, a line spacing of 2 means the line spacing is two times the default line spacing. The default line spacing is the line spacing as specified in the font used.
bgColor	The background color of the block in hex format.
valign	The vertical alignment of sub-blocks. This is for blocks that contain sub-blocks. Supported values are baseline, top, bottom, middle and absmiddle.

is the underline position of text. This is normal method of aligning text, and is the default in CDML. For images or blocks that are rotated, the baseline is the same as the bottom.

The value top means the top line of sub-blocks should align with the top line of the block.

The value bottom means the bottom line of sub-blocks should align with the bottom line of the block.

The value middle means the middle line of sub-blocks should align with the the middle line of the block. The middle line is the middle position between the top line and the baseline.

The value absmiddle means the absolute middle line of sub-blocks should align with the absolute middle line of the block. The absolute middle line is the middle position between the top line and the bottom line.

halign	The horizontal alignment of lines. This is for blocks that contain multiple lines. Supported values are left, center and right.

The value left means the left border of each line should align with the left border of the block. This is the default.

The value center means the horizontal center of each line should align with the horizontal center of the block.

The value right means the right border of each line should align with the right border of the block.

angle Rotate the content of the block by an angle. The angle is specified in degrees in counter-clockwise direction.

20.0.27 ChartDirector: Parameter Substitution and Formatting

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Parameter Substitution and Formatting
Notes: ChartDirector charts often contain a lot of text strings. For example, sector labels in pie charts, axis labels for x and y axes, data labels for the data points, HTML image maps, etc, are all text strings.

ChartDirector uses parameter substitution to allow you to configure precisely the information contained in the text and their format.
Format Strings
In parameter substitution, format strings are used to specify the entities to be include into labels and how to format numbers and dates.

For example, when drawing a pie chart with side label layout, the default sector label format string is:

" { label } ({ percent } %)"

When the sector label is actually drawn, ChartDirector will replace " { label } " with the sector name, and " { percent } " with the sector percentage. So the above label format will result is a sector label similar to "ABC (34.56%)".

You may change the sector label format by changing the format string. For example, you may change it to:

" { label } : US$ { value | 2 } K ({ percent } %)"

The sector label will then become something like "ABC: US$ 123.00 (34.56%)".

In general, in ChartDirector parameter substitution, parameters enclosed by curly brackets will be substituted with their actual values when creating the texts.

For parameters that are numbers or dates/times, ChartDirector supports a special syntax in parameter substitution to allow formatting for these values. Please refer to the Number Formatting and Date/Time Formatting sections below for details.

Parameter Expressions

ChartDirector supports numeric expressions in format strings. They are denoted by enclosing the expression with curly brackets and using "=" as the first character. For example:

"USD { value } (Euro { = { value } *0.9 })"

In the above, " { value } " will be substituted with the actual value of the sector. The expression " { = { value } *0.9 } " will be substituted with the actual value of the sector multiplied by 0.9.

ChartDirector parameter expressions support operators "+", "-", "*", "/", "%" (modulo) and "^" (exponentiation). Operators "*", "/", "%", "^" is computed first, followed by "+" and "-". Operators of the same precedence are computed from left to right). Parenthesis "(" and ")" can be used to change the computation order.

Parameters for Pie Charts

The following table describes the parameters available for pie charts.

Parameter	Description
sector	The sector number. The first sector is 0, while the nth sector is (n-1).
dataSet	Same as { sector } . See above.
label	The text label of the sector.
dataSetName	Same as { label } . See above.
value	The data value of the sector.
percent	The percentage value of the sector.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using BaseChart.addExtraField or BaseChart.addExtraField2.

Parameters for All XY Chart Layers

The followings are parameters that are apply to all XY Chart layers in general. Some layer types may have additional parameters (see below).

Note that certain parameters are inapplicable in some context. For example, when specifying the aggregate label of a stacked bar chart, the { dataSetName } parameter is inapplicable. It is because a stacked bar is composed of multiple data sets. It does not belong to any particular data set and hence does not have a data set name.

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for Line Layers
The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Trend Layers
The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Box-Whisker Layers
The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for HLOC and CandleStick Layers
The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Vector Layers
The followings are parameters that are in additional to the parameters for all XY Chart layers.

Parameters for All Polar Layers
The followings are parameters that are apply to all Polar Chart layers in general. Some layer types may have additional parameters (see below).

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for PolarVector Layers
The followings are parameters that are in additional to the parameters for all Polar Chart layers.

Parameters for Axis
The following table describes the parameters available for pie charts.

Number Formatting
For parameters that are numbers, ChartDirector supports a number of formatting options in parameter substitution.

For example, if you want a numeric field { value } to have a precision of two digits to the right of the decimal point, use ',' (comma) as the thousand separator, and use '.' (dot) as the decimal point, and you may use { value | 2,. } . The number 123456.789 will then be displayed as 123,456.79.

For numbers, the formatting options are specified using the following syntax:

{ [param] | [a] [b] [c] [d] }

where:

If this field starts with "E" or "e", followed by a number, it means formatting the value using scientific notation with the specified number of decimal places. If the "E" or "e" is not followed by a number, 3 is assumed.

For example, { value | E4 } will format the value 10.3 to 1.0300E+1, and { value | e4 } will format the same value to 1.0300e+1.

If this field starts with "G" or "g", followed by a number, it means formatting the value using the scientific notation only if the value is large and requires more than the specified number of digits, or the value is less than 0.001. If scientific notation is used, the number following "G" or "g" also specifies the number of significant digits to use. If the "G" or "g" is not followed by a number, 4 is assumed.

For example, consider the format string { value | G4 } . The value 10 will be formatted to 10. The value 100000 will be formatted to 1.000E+5. Similarly, for { value | g4 } , the value 10 will be formatted to 10, while the value 100000 will be formatted to 1.000e+5.

If you skip this argument, ChartDirector will display the exact value using at most 6 decimal places.

You may skip [b] [c] [d] . In this case, the default will be used.

Date/Time Formatting
For parameters that are dates/times, the formatting options can be specified using the following syntax:

{ [param] | [datetime_format_string] }

where [datetime_format_string] must start with an english character (A-Z or a-z) that is not "G", "g", "E" or "e", and may contain any characters except ' } '. (If it starts with "G", "g", "E" or "e", it will be considered as a number format string.)

Certain characters are substituted according to the following table. Characters that are not substituted will be copied to the output.

For example, a parameter substitution format of { value | mm-dd-yyyy } will display a date as something similar to 09-15-2002. A format of { value | dd/mm/yy hh:nn:ss a } will display a date as something similar to 15/09/02 03:04:05 pm.

If you want to include characters in the format string without substitution, you may enclose the characters in single or double quotes.

For example, the format { value | mmm '<*color=dd0000*>'yyyy } will display a date as something like Jan <*color=dd0000*>2005 (the <*color=dd0000*>is a CDML tag to specify red text color). Note that the <*color=dd0000*>tag is copied directly without substitution, even it contains "dd" which normally will be substituted with the day of month.
Escaping URL/HTML/CDML characters
Parameter substitution is often used to create HTML image maps. In HTML, some characters has special meanings and cannot be used reliably. For example, the '>' is used to represent the end of an HTML tag.

Furthermore, if the field happens to be used as an URL, characters such as '?', '&' and '+' also have special meanings.

By default, ChartDirector will escape template fields used in URL and query parameters when generating image maps. It will modify URL special characters to the URL escape format "%XX" (eg. "?" will become "%3F"). After that, it will modify HTML special characters to the HTML escape format "&#nn;" (eg. ">" will become ">".). Similarly, it will escape other attributes in the image map using HTML escape format (but not URL escape format).

In addition to escaping HTML and URL special characters, ChartDirector will also remove CDML fields in creating image maps. It is because CDML is only interpreted in ChartDirector, should not be useful outside of ChartDirector (such as in browser tool tips).

In some cases, you may not want ChartDirector to escape the special characters. For example, if the parameters have already been escaped before passing to ChartDirector, you may want to disable ChartDirector from escaping them again.

ChartDirector supports the following special fields to control the escape methods - " { escape_url } ", " { noescape_url } ", " { escape_html } ", " { noescape_html } ", " { escape_cdml } " and { noescape_cdml } ". These fields enable/disable the escape methods used in the template fields that follow them.

20.0.28 ChartDirector: Shape Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Shape Specification
Notes: Several ChartDirector API accept shape specification as arguments. For example, BarLayer.setBarShape and BarLayer.setBarShape2 can be used to specify shapes of bars in bar charts, while DataSet.setDataSymbol, DataSet.setDataSymbol4, PolarLayer.setDataSymbol and PolarLayer.setDataSymbol4 can be used to specify shapes for data symbols.

Note that in addition to shapes, in many cases ChartDirector also accepts images or custom draw objects for data representation. For example, see DataSet.setDataSymbol2, DataSet.setDataSymbol3, PolarLayer.setDataSymbol2 and PolarLayer.setDataSymbol3.

Built-In Shapes

Built-in shapes are specified as integers. The integers can be explicit constants, or can be generated by a ChartDirector method for parameterized shapes. For example, a circle is represented by an explicit constant CircleShape (=7). On the other hand, the number representing a polygon depends on the number of sides the polygon has, so it is generated by using the PolygonShape method, passing in the number of sides as argument.

The following table illustrates the various ChartDirector shapes:

Custom Shapes

In ChartDirector, custom shapes are specified as an array of integers x0, y0, x1, y1, x2, y2 ... representing the coordinates of the vertices of the custom polygonal shape.

The polygon should be defined with a bounding square of 1000 x 1000 units, in which the x-axis is from -500 to 500 going from left to right, and the y-axis is from 0 to 1000 going from bottom to top.

ChartDirector will automatically scale the polygon so that 1000 units will become to the pixel size as requested by the various ChartDirector API.

As an example, the shape of the standard diamond shape in ChartDirector is represented as an array with 8 numbers:

0, 0, 500, 500, 0, 1000, -500, 500

20.0.29 Copy styled text?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: How to quickly copy styled text from one textarea to another?
Example:

```
#if TargetWin32 then
TextArea1.WinRTFDataMBS = TextArea2.WinRTFDataMBS
#elseif TargetMacOS then
TextArea1.NSTextViewMBS.textStorage.setAttributedString TextArea2.NSTextViewMBS.textStorage
#else
TextArea1.StyledText = TextArea2.StyledText
#endif
```

Notes: The code above uses special plugin functions on Mac and Windows and falls back to framework for Linux.

20.0.30 Do you have code to validate a credit card number?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can check the checksum to tell if a credit card number is not valid.
Example:

```
Dim strNumber As String
Dim nLength as Integer
Dim nValue as Integer
Dim nChecksum as Integer
Dim nIndex as Integer

strNumber = EditField1.Text
nLength = Len(strNumber)
nChecksum = 0

For nIndex = 0 To nLength - 2
nValue = Val(Mid(strNumber, nLength - (nIndex + 1), 1)) * (2 - (nIndex Mod 2))
If nValue <10 Then
nChecksum = nChecksum + nValue
Else
nChecksum = nChecksum + (nValue - 9)
End If
Next

If Val(Mid(strNumber, Len(strNumber), 1)) = (10 - (nChecksum Mod 10)) Mod 10 Then
MsgBox("The credit card number looks valid")
Else
MsgBox("The credit card number is invalid")
End IF
```

Notes: Here's some code that will validate the checksum for a credit card. It works for Visa, MasterCard, American Express and Discover. Not sure about others, but I imagine they use the same basic algorithm. Of course, this doesn't actually mean that the credit card is valid, it's only useful for helping the user catch typos.

The above code doesn't have any error checking and it expects that the credit card number will be entered without spaces, dashes or any other non-numeric characters. Addressing those issues will be an exercise left to the reader. :)

(From Mike Stefanik)

20.0.31 Do you have plugins for X-Rite EyeOne, eXact or i1Pro?

Plugin Version: all.

Answer: Our EyeOne plugin is available on request for licensees of the X-Rite SDKs.
Notes: Please first go to X-Rite and get a SDK license.
Than we can talk about the plugin.

20.0.32 Does SQL Plugin handle stored procedures with multiple result sets?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, the plugin can work with multiple recordsets.
Notes: You need to use SQLCommandMBS class. When you get back results, you use FetchNext to walk over all records in the first result set. Than you simply start again with FetchNext to get the second record set.
Even the RecordSet functions should work, just use them twice to get all records from both record sets.

20.0.33 Does the plugin home home?

Plugin Version: all, Platform: macOS.

Answer: Yes, we like to know who is using the plugin, so the plugin may contact our server.
Example:

none.

Notes: Please note that this does not affect your users as the plugin will only do this in the IDE and the relevant plugin part is never included in your applications.

The plugin if used for some hours, does contact our server to provide statistical data about Xojo version and OS versions. This way we know what versions are used. We can return the version number of the current plugin which may be visible in future versions somehow. And we transmit partial licenses data so we can track use of illegal license keys.

If you do not like to have this, you can block Xojo IDE from contacting our website via your Firewall.
Blocking the transfer will not disable the plugin or change the features.
Or contact us for a plugin version which explicitly does not contain this feature.

20.0.34 folderitem.absolutepath is limited to 255 chars. How can I get longer ones?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Paths on a Mac are not unique, so use them only to display them to the user.
Example:

```
Function AbsolutePath(f as FolderItem) As String
Dim s as string
Dim nf as FolderItem
nf = f
s = ""
while nf<>nil
s = nf.name + ":" + s
nf = nf.parent
wend
Return s
End Function
```

20.0.35 Has anyone played round with using CoreImage to do things like add dissolve transitions say when changing from one tab to another within a window?

Platform: macOS.

Answer: This code implements animations for a tabpanel change:
Example:

```
// in a tabpanel.change event:

dim r as CGSTransitionRequestMBS
dim co as new CGSConnectionMBS
dim cw as CGSWindowMBS
dim ct as CGSTransitionMBS
static OldTab as Integer

cw=co.CGSWindow(window1)
If cw = Nil Then
return // 10.3...
End If
r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSFlip
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw
```

```
// watch the value of the clicked tab versus the last tab
if tabpanel1.Value=0 or tabpanel1.Value <OldTab then
r.TransitionOption=r.CGSLeft
ct=co.NewTransition(r)
if ct<>Nil then
Refresh
ct.Invoke(1)
ct.Wait(1)
ct.Release
else
MsgBox "Error creating the transition."
end if
else
r.TransitionOption=r.CGSRight
ct=co.NewTransition(r)
if ct<>Nil then
Refresh
ct.Invoke(1)
ct.Wait(1)
ct.Release
else
MsgBox "Error creating the transition."
end if
end if
// Keep track of the last tab clicked
OldTab = tabpanel1.Value
```

Notes: See CGS* classes for more details.

20.0.36 How about Plugin support for older OS X?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We support in general Mac OS X 10.5 and newer.
Notes: All the 64-bit plugins on Mac require OS X 10.7.
Intel 32-bit plugins on Mac require OS X 10.5 or newer.

Currently the ChartDirector 6, GraphicsMagick and GameKit plugins requires Mac OS X 10.6.
Also for SQL Plugin the built in SQLite library requires 10.6.

20.0.37 How can I detect whether an Intel CPU is a 64bit CPU?

Plugin Version: all.

Answer: Look on the CPU family returned by sysctl:
Example:

```
Function is64bit() As Boolean

#if TargetLittleEndian

dim m as MemoryBlock = NewMemoryBlock(8)
dim family as Integer
dim s as string

m=SystemControlNameToMIBMBS("hw.cpufamily")
m=SystemControlMBS(m)

if m<>nil then
m.LittleEndian=True
family=m.Long(0)

const CPUFAMILY_INTEL_6_14 = &h73d67300 //* "Intel Core Solo" and "Intel Core Duo" (32-bit Pentium-M with SSE3) */
const CPUFAMILY_INTEL_6_15 = &h426f69ef //* "Intel Core 2 Duo" */
const CPUFAMILY_INTEL_6_23 = &h78ea4fbc //* Penryn */
const CPUFAMILY_INTEL_6_26 = &h6b5a4cd2 //* Nehalem */

Select case family
case CPUFAMILY_INTEL_6_14
Return false
case CPUFAMILY_INTEL_6_15
Return true
case CPUFAMILY_INTEL_6_23
Return true
case CPUFAMILY_INTEL_6_26
Return true
// newer CPUs may be missing here
end Select

end if

#endif

Return false

Exception
Return false
End Function
```

Notes: This code is written for Mac OS X where you only have a limited number of possible CPUs.

20.0.38 How can I disable the close box of a window on Windows?

Plugin Version: all, Platform: Windows.

Answer: The following code will remove the close item from the system menu of the window.
Example:

```
#if TargetWin32 then
Declare Function GetSystemMenu Lib "user32" (hwnd as Integer, bRevert as Integer) as Integer
Declare Function RemoveMenu Lib "user32" (hMenu as Integer, nPosition as Integer, wFlags as Integer) as
Integer
Dim hSysMenu as Integer
hSysMenu = GetSystemMenu(me.WinHWND, 0)
RemoveMenu hSysMenu, &HF060, &H0
#endif
```

Notes: The window may not be updated directly.

20.0.39 How can I get all the environment variables from Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetWin32
declare function GetEnvironmentStrings Lib "kernel32" () as ptr
dim m as memoryBlock
dim n as Integer

m=GetEnvironmentStrings()

n=0
do
msgBox m.cstring(n)
while m.byte(n)<>0
n=n+1
wend
n=n+1
```

```
loop until m.byte(n)=0
#endif
```

Notes: The MBS Plugin has an EnvironmentMBS class for this.

20.0.40 How can i get similar behavior to Roxio Toast or iTunes where clicking a 'burn' button allows the next inserted blank CD-R to bypass the Finder and be accepted by my application?

Plugin Version: all, Platform: macOS.

Answer: You need to get a media reservation.
Example:

```
dim d as DRDeviceMBS // get a device
d.AcquireMediaReservation
```

Notes: Use the plugin function AcquireMediaReservation and later release it using ReleaseMediaReservation.
See plugin examples on how to use it and check Apples DiscRecording framework documentation for more details.

20.0.41 How can I get text from a PDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Crossplatform you can use DynaPDF Pro.
Notes: On Mac OS X you can also use PDFKit for the same job.
While DynaPDF Pro gives you each bit of text with rotation, font information and encoding details, PDFKit gives you only the text string for a PDF page.

20.0.42 How can I get text from a Word Document?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: to get the text string from a doc file, use the NSAttributedStringMBS class.
Notes: The NSAttributedStringMBS class is Mac OS X only and we have currently no solution for Windows or Linux.

Use the NSAttributedStringMBS.initWithDocFormat(data as string) as boolean method.

20.0.43 How can I get the item string for a given file creator?

Plugin Version: all.

Answer: Try this function:
Example:

```
Sub pullNativeDocs(aCREA As string)
Dim result as Integer
Dim m, k as memoryBlock
Dim f as folderItem
Dim newType as string
Dim anIcon As picture
Dim ofs as Integer

Declare Function GetFileTypesThatAppCanNativelyOpen Lib "Carbon" (appVRefNumHint as Short, appSignature as OSType, nativeTypes as Ptr) as Short Inline68K("701CABFC")
Declare Function GetDocumentKindString Lib "Carbon" (docVRefNum as Short, docType as OSType, docCreator as OSType, kindString as ptr) as Short Inline68K("7016ABFC")

listBox1.deleteAllRows

m = newMemoryBlock(1024)
result = GetFileTypesThatAppCanNativelyOpen(Volume(0).MacVRefNum, aCREA, m)
if result <>0 then
listBox1.addRow "<Not found.>"
return
end if

do
if m.byte(ofs*4) = 0 then
exit
else
newType = m.OSTypeMBS(ofs*4)
listBox1.addRow newType
k = newMemoryBlock(64)
result = GetDocumentKindString(Volume(0).MacVRefNum, newType, aCREA, k)
if result = 0 then
listBox1.cell(ofs,1) = k.pString(0)
ofs = ofs + 1
else
listBox1.cell(ofs,1) = "(unknown)"
end if

end if
```

```
loop

End Sub
```

Notes: Change ”Translation” to ”CarbonLib” for Mac OS X.

20.0.44 How can I launch an app using it’s creator code?

Plugin Version: all, Platform: macOS.

Answer: Send an AppleEvent ”odoc” with the creator code to the Finder (”MACS”):
Example:

```
Function LaunchByCreator(C As String) As Boolean
Dim A As AppleEvent
A = NewAppleEvent(”aevt”,”odoc”,”MACS”)
A.ObjectSpecifierParam(”—-”) = GetUniqueIDObjectDescriptor(”appf”,nil,C)
return A.Send
End Function
```

20.0.45 How can I learn what shared libraries are required by a plugin on Linux?

Plugin Version: all, Platform: macOS.

Answer: Please use the ldd command in the terminal.
Notes: You build an app on any platform, but for Linux.
For the resulting .so files in the libs folder, you can run the ldd command with the library path as parameter.
It shows you references lib files and you can make sure you have those installed.

This is a sample run of our graphicsmagick plugin:

```
cs@Ubuntu32:
textasciitilde /MeinProgramm/MeinProgramm Libs$ ldd libMBSGraphicsMagickPlugin17744.so
linux-gate.so.1 =>(0xb76ee000)
libdl.so.2 =>/lib/i386-linux-gnu/libdl.so.2 (0xb6f0e000)
libgtk-x11-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgtk-x11-2.0.so.0 (0xb6aa6000)
libpthread.so.0 =>/lib/i386-linux-gnu/libpthread.so.0 (0xb6a8a000)
libstdc++.so.6 =>/usr/lib/i386-linux-gnu/libstdc++.so.6 (0xb69a5000)
libm.so.6 =>/lib/i386-linux-gnu/libm.so.6 (0xb6979000)
libgcc_s.so.1 =>/lib/i386-linux-gnu/libgcc_s.so.1 (0xb695b000)
libc.so.6 =>/lib/i386-linux-gnu/libc.so.6 (0xb67b1000)
```

```
/lib/ld-linux.so.2 (0xb76ef000)
libgdk-x11-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgdk-x11-2.0.so.0 (0xb6701000)
libpangocairo-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpangocairo-1.0.so.0 (0xb66f4000)
libX11.so.6 =>/usr/lib/i386-linux-gnu/libX11.so.6 (0xb65c0000)
libXfixes.so.3 =>/usr/lib/i386-linux-gnu/libXfixes.so.3 (0xb65ba000)
libatk-1.0.so.0 =>/usr/lib/i386-linux-gnu/libatk-1.0.so.0 (0xb659a000)
libcairo.so.2 =>/usr/lib/i386-linux-gnu/libcairo.so.2 (0xb64ce000)
libgdk_pixbuf-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgdk_pixbuf-2.0.so.0 (0xb64ad000)
libgio-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgio-2.0.so.0 (0xb6356000)
libpangoft2-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpangoft2-1.0.so.0 (0xb632a000)
libpango-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpango-1.0.so.0 (0xb62e0000)
libfontconfig.so.1 =>/usr/lib/i386-linux-gnu/libfontconfig.so.1 (0xb62ab000)
libgobject-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgobject-2.0.so.0 (0xb625c000)
libglib-2.0.so.0 =>/lib/i386-linux-gnu/libglib-2.0.so.0 (0xb6163000)
libXext.so.6 =>/usr/lib/i386-linux-gnu/libXext.so.6 (0xb6151000)
libXrender.so.1 =>/usr/lib/i386-linux-gnu/libXrender.so.1 (0xb6147000)
libXinerama.so.1 =>/usr/lib/i386-linux-gnu/libXinerama.so.1 (0xb6142000)
libXi.so.6 =>/usr/lib/i386-linux-gnu/libXi.so.6 (0xb6132000)
libXrandr.so.2 =>/usr/lib/i386-linux-gnu/libXrandr.so.2 (0xb6129000)
libXcursor.so.1 =>/usr/lib/i386-linux-gnu/libXcursor.so.1 (0xb611e000)
libXcomposite.so.1 =>/usr/lib/i386-linux-gnu/libXcomposite.so.1 (0xb611a000)
libXdamage.so.1 =>/usr/lib/i386-linux-gnu/libXdamage.so.1 (0xb6115000)
libfreetype.so.6 =>/usr/lib/i386-linux-gnu/libfreetype.so.6 (0xb607b000)
libxcb.so.1 =>/usr/lib/i386-linux-gnu/libxcb.so.1 (0xb605a000)
libpixman-1.so.0 =>/usr/lib/i386-linux-gnu/libpixman-1.so.0 (0xb5fc2000)
libpng12.so.0 =>/lib/i386-linux-gnu/libpng12.so.0 (0xb5f98000)
libxcb-shm.so.0 =>/usr/lib/i386-linux-gnu/libxcb-shm.so.0 (0xb5f93000)
libxcb-render.so.0 =>/usr/lib/i386-linux-gnu/libxcb-render.so.0 (0xb5f89000)
libz.so.1 =>/lib/i386-linux-gnu/libz.so.1 (0xb5f73000)
libgmodule-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgmodule-2.0.so.0 (0xb5f6e000)
libselinux.so.1 =>/lib/i386-linux-gnu/libselinux.so.1 (0xb5f4f000)
libresolv.so.2 =>/lib/i386-linux-gnu/libresolv.so.2 (0xb5f36000)
libexpat.so.1 =>/lib/i386-linux-gnu/libexpat.so.1 (0xb5f0c000)
libffi.so.6 =>/usr/lib/i386-linux-gnu/libffi.so.6 (0xb5f05000)
libpcre.so.3 =>/lib/i386-linux-gnu/libpcre.so.3 (0xb5ec9000)
librt.so.1 =>/lib/i386-linux-gnu/librt.so.1 (0xb5ec0000)
libXau.so.6 =>/usr/lib/i386-linux-gnu/libXau.so.6 (0xb5ebb000)
libXdmcp.so.6 =>/usr/lib/i386-linux-gnu/libXdmcp.so.6 (0xb5eb4000)
cs@Ubuntu32:
textasciitilde /MeinProgramm/MeinProgramm Libs$
```

As you see all library have been found and their load address is printed behind the na,e.
If a library is missing, you usually see the address missing there or being zero.

20.0.46 How can I validate an email address?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
Dim re As RegEx
re = New RegEx
Dim rm As RegExMatch

re.SearchPattern = " [ a-z0-9!#$ %&'*+/=?^_' { | }
textasciitilde - ] +(?:\. [ a-z0-9!#$ %&'*+/=?^_' { | }
textasciitilde - ] +)*@(?: [ a-z0-9 ] (?: [ a-z0-9- ] * [ a-z0-9 ] )?\.)+ [ a-z0-9 ] (?: [ a-z0-9- ] * [ a-z0-9 ] )?"
rm = re.Search(editField1.Text)

if rm = Nil Then
StaticText2.text = editField1.Text + " not valid email"
Else
StaticText2.Text = editField1.Text + " is valid"
End if
```

Notes: Adapted from:
http://www.regular-expressions.info/email.html

20.0.47 How do I decode correctly an email subject?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following code can be used to decode an email subject including several encodings including Base 64.
Example:

```
dim src as string // input

dim theRegex as Regex
dim theRegexMatch as RegexMatch
dim result, infoCharset, encodedPart as string
dim theStart as Integer

if instr(src, "=?") >0 then
theRegex = new Regex
theRegex.Options.Greedy = false
theRegex.searchPattern = "(.*)=\?(.+)\?(Q | B)\?(.+)\?="
theRegexMatch = theRegex.search(src)
```

```
while theRegexMatch <>nil
theStart = theRegexMatch.subExpressionStartB(0) + len(theRegexMatch.subExpressionString(0))

result = result + theRegexMatch.subExpressionString(1)
infoCharset = theRegexMatch.subExpressionString(2)
encodedPart = theRegexMatch.subExpressionString(4)
if theRegexMatch.subExpressionString(3) = "B" then
encodedPart = DecodeBase64(encodedPart)
elseif theRegexMatch.subExpressionString(3) = "Q" then
encodedPart = DecodeQuotedPrintable(encodedPart)
end if
if right(result, 1) = " " then
result = mid(result, 1, len(result)-1)
end if
encodedPart = encodedPart.DefineEncoding(GetInternetTextEncoding(infoCharset))
result = result + encodedPart

theRegex.SearchStartPosition = theStart
theRegexMatch = theRegex.search()
wend

result = result + mid(src, theStart+1)

else
result = src
end if
// theRegexMatch = theRegex.search

msgbox result
```

Notes: May not look nice depending on the controls used.
This is no longer needed when using MimeEmailMBS class which decodes for you.

20.0.48 How do I enable/disable a single tab in a tabpanel?

Plugin Version: all, Platform: macOS.

Answer: Use the TabpanelEnabledMBS method.
Example:

```
TabpanelEnabledMBS(tabpanel1, 1, false)
```

Notes: Use Carbon for MachO and CarbonLib for Mac Carbon and AppearanceLib for Mac OS Classic as

library.
For Cocoa, please use enabled property of NSTabViewItemMBS class.

20.0.49 How do I find the root volume for a file?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this function:
Example:

```
Function GetRootVolume(f as FolderItem) as FolderItem
dim root, dum as folderItem
if f <>nil then
root = f // f might be the volume
do
dum = root.parent
if dum <>nil then
root = dum
end if
loop until dum = nil
return root
end if
End Function
```

20.0.50 How do I get the current languages list?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim p as new CFPreferencesMBS
dim a as CFArrayMBS
dim s as CFStringMBS
dim o as CFObjectMBS
dim sa(-1) as string

o=p.CopyAppValue("AppleLanguages",".GlobalPreferences")

if o<>Nil then
a=CFArrayMBS(o)

dim i,c as Integer
```

```
c=a.Count-1
for i=0 to c
o=a.Item(i)

if o isa CFStringMBS then
s=CFStringMBS(o)
sa.Append s.str
end if
next
end if

MsgBox Join(sa,EndOfLine)
```

Notes: On Mac OS X you can get the list of current languages like this list:

```
de
en
ja
fr
es
it
pt
pt-PT
nl
sv
nb
da
fi
ru
pl
zh-Hans
zh-Hant
ko
```

Which has German (de) on the top for a German user.
This code has been tested on Mac OS X 10.5 only.

20.0.51 How do I get the Mac OS Version?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim i as Integer
if system.gestalt("sysv", i) then
//do this in an 'If' in case you don't get any value back at all and system.gestalt returns boolean
if i = &h750 then //If OS is 7.5
//do stuff
elseif i = &h761 then //If OS is 7.6.1
//do stuff
end if
end if
```

Notes: The MBS Plugin has a function SystemInformationMBS.OSVersionString for this.

20.0.52 How do I get the printer name?

Plugin Version: all.

Answer: For Mac OS Classic see the code below and for Mac OS X use the Carbon Print Manager Classes from the MBS Plugin.
Example:

```
dim s as String
dim i as Integer

s=app.ResourceFork.GetResource("STR ",-8192)
if s<>"" then
i=ascb(leftb(s,1))
s=mid(s,2,i)

MsgBox s
end if
```

Notes: A note from Craig Hoyt:

After looking at your example I had a little deja-vu experience. Several years ago I played around with this same code if FutureBasic. I discovered that it did not and still doesn't provide the 'Printer Name', it does return the print driver name. If it returns 'LaserWriter 8' as the print driver you can look into this file and get the 'PAPA' resource #-8192 to get the actual Printer Name. Unfortunately this does not hold true for other printers. My Epson and HP Printers (the Epson has an Ethernet Card and the HP is USB) do not provide this info in their drivers. As far as I can tell it only returns the name by polling the printer itself.

20.0.53 How do I make a metal window if RB does not allow me this?

Plugin Version: all, Platform: macOS.

Answer: The following declare turns any window on Mac OS X 10.2 or newer into a metal one.
Example:

```
declare sub ChangeWindowAttributes lib "Carbon" (win as windowptr, a as Integer, b as Integer)

ChangeWindowAttributes window1,256,0
```

Notes: May not look nice depending on the controls used.

20.0.54 How do I make a smooth color transition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

I'd like to show in a report some bars, which start with color A
and end with color B.

The color change should be very smooth.

My problem: If I would start from 255,0,0 and end by 0,0,0, I would have
255 different colors. If the bars are longer than 255 pixels, would
this look nice?
Example:

```
// Window.Paint:
Sub Paint(g As Graphics)
dim w,w1,x,p as Integer
dim c1,c2,c as color
dim p1,p2 as Double

c1=rgb(255,0,0) // start color
c2=rgb(0,255,0) // end color

w=g.Width
w1=w-1

for x=0 to w1
p1=x/w1
p2=1.0-p1
```

```
c=rgb(c1.red*p1+c2.red*p2, c1.green*p1+c2.green*p2, c1.blue*p1+c2.blue*p2)

g.ForeColor=c
g.DrawLine x,0,x,g.Height

next
End Sub
```

Notes:

Try the code above in a window paint event handler.

20.0.55 How do I read the applications in the dock app?

Plugin Version: all, Platform: macOS.

Answer: Use CFPreferencesMBS class like in this example:
Example:

```
// Reads file names from persistent dock applications and puts them into the list

dim pref as new CFPreferencesMBS

dim persistentapps as CFStringMBS = NewCFStringMBS("persistent-apps")
dim ApplicationID as CFStringMBS = NewCFStringMBS("com.apple.dock")
dim tiledata as CFStringMBS = NewCFStringMBS("tile-data")
dim filelabel as CFStringMBS = NewCFStringMBS("file-label")

// get the array of persistent applications from dock preferences
dim o as CFObjectMBS = pref.CopyValue(persistentapps, ApplicationID, pref.kCFPreferencesCurrentUser,
pref.kCFPreferencesAnyHost)

if o isa CFArrayMBS then
dim a as CFArrayMBS = CFArrayMBS(o)

// walk over all items in array
dim c as Integer = a.Count-1
for i as Integer = 0 to c

// get dictionary describing item
o = a.Item(i)

if o isa CFDictionaryMBS then
dim d as CFDictionaryMBS = CFDictionaryMBS(o)
```

```
// and pick tile data dictionary
o = d.Value(tiledata)
if o isa CFDictionaryMBS then
d = CFDictionaryMBS(o)

// and pick there the file label
o = d.Value(filelabel)
if o isa CFStringMBS then
// and display it
dim name as string = CFStringMBS(o).str
List.AddRow name
end if
end if
end if

next

else
MsgBox "Failed to read dock preferences."
end if
```

Notes: You can use the CFPreferencesMBS.SetValue to change a value and CFPreferencesMBS.Synchronize to write the values to disc. You may need to restart the Dock.app if you modified things.

20.0.56 How do I truncate a file?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In a binarystream you can set the length property to truncate.

20.0.57 How do update a Finder's windows after changing some files?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim f as folderitem // some file
dim ae as appleevent
ae=newappleevent("fndr","fupd","MACS")
ae.folderitemparam("—-")=f
if not ae.send then
//something went wrong
```

```
end if
```

Notes: The folderitem.finderupdate from the MBS Plugin does something like this.

20.0.58 How to access a USB device directly?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: First, it depends on the device.
Notes: Some devices can be talked directly from user mode code, but some require a kernel driver.

For some devices you can use plugins to access them like:

- Audio and Video sources using the QTGrabberClassMBS
- Mass storage devices using the folderitem class.
- Serial devices using the System.SerialPort function.
- HID USB devices can be used with MacHIDMBS, WinHIDMBS or LinuxHIDInterface class.
- Any USB device may be used with MacUSBMBS or WinUSBMBS classes.

In general it is always the best to take the most high level access to have others do the work for the details.

20.0.59 How to add icon to file on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use FolderItem.AddCustomIcon or NSWorkspaceMBS.setIcon functions.
Notes: Please close any open stream for the file you want to add an icon.

20.0.60 How to ask the Mac for the Name of the Machine?

Plugin Version: all, Platform: macOS.

Answer: Using Apple Events you can use this code:
Example:

```
Function Computername() As string
```

```
dim theEvent as AppleEvent
dim err as boolean

theEvent = newAppleEvent("mchn","getd","MACS")

err = theEvent.send

return theevent.ReplyString

End Function
```

Notes: Code above is for Mac OS 9!
Also the MBS Plugin has a function for this which may be faster and work also on Macs without Filesharing (which handles this event).

20.0.61 How to automatically enable retina in my apps?

Plugin Version: all, Platform: macOS.

Answer: You can run a build script on each build with this code:
Example:

```
Dim App As String = CurrentBuildLocation + "/" + CurrentBuildAppName + ".app"
Call DoShellCommand("/usr/bin/defaults write " + App + "/Contents/Info ""NSHighResolutionCapable"" YES")
```

Notes: This will set the NSHighResolutionCapable flag to YES.

20.0.62 How to avoid leaks with Cocoa functions?

Plugin Version: all, Platform: macOS.

Answer: You can try this code on Mac OS X:
Example:

```
// in a Timer Action event:
Sub Action()
static LastPool as NSAutoreleasePoolMBS = nil
static CurrentPool as NSAutoreleasePoolMBS = nil

LastPool = CurrentPool
CurrentPool = new NSAutoreleasePoolMBS
```

```
End Sub
```

Notes: With Xojo 2009r4 the code above should not be needed as Xojo runtime does automatically handle the NSAutoreleasePools for you. For older Xojo versions you need to use code with a timer with the action event above to avoid memory leaks.

Please do not use Xojo 2009r4 and newer with plugins before version 9.5. You can get crashes there which typically show a line with a objc_msgSend call.

20.0.63 How to avoid trouble connecting to oracle database with SQL Plugin?

Plugin Version: all, Platform: macOS.

Answer: For oracle the most important thing is to point the plugin to the libraries from oracle.
Notes: In environment variables, the paths like ORACLE_HOME must be defined.
On Mac OS X you also need to define DYLD_LIBRARY_PATH to point to the dylib files from oracle.

For that you need to modify /etc/launchd.conf for Mac OS X 10.8 and newer.
In older versions those variables in .MacOSX/environment.plist file in user's home.

Another way for the case you bundle things inside your app is to use the LSEnvironment key in info.plist.
In info.plist it looks like this:

```
<key>LSEnvironment</key>
<dict>
<key>test</key>
<string>Hello World</string>
</dict>
```

20.0.64 How to avoid ___NSAutoreleaseNoPool console messages in threads?

Plugin Version: all, Platform: macOS.

Answer: You need to use your own NSAutoreleasePool on a thread like this:
Example:

```
sub MyThread.run
dim pool as new NSAutoreleasePoolMBS
// do work here

pool=nil
```

```
end sub
```

Notes: For more details read here:
http://developer.apple.com/mac/library/documentation/Cocoa/Reference/Foundation/Classes/NSAutoreleasePool_Class/Reference/Reference.html

20.0.65 How to bring app to front?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac you can use this code:
Example:

```
// First way:
app.FrontMostMBS = true

// second way:
dim p as new ProcessMBS
p.GetCurrentProcess
p.FrontProcess = true

// third way:
NSApplicationMBS.sharedApplication.activateIgnoringOtherApps(true)

// for Windows:
RemoteControlMBS.WinBringWindowToTop
```

Notes: This will bring a Mac app to the front layer.

20.0.66 How to bring my application to front?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code ttxt) to the frontmost application:
Example:

```
Dim A As AppleEvent
A = NewAppleEvent("misc","actv","")
If Not A.Send then
Beep
end if
```

Notes: (Code is Mac only)

20.0.67 How to catch Control-C on Mac or Linux in a console app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use SignalHandlerMBS class for this.
Example:

```
// watch for Control-C on Mac
call SignalHandlerMBS.SetFlagHandler(2)

dim ende as boolean = false
do
if SignalHandlerMBS.IsFlagSet(2) then
Print "Flag 2 set. Existing..."
ende = true
end if

DoEvents 1
loop until ende
```

Notes: The signal is catched, a flag is set and you can ask later in your normal application flow for the result.

20.0.68 How to change name of application menu?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Use this code to change the application menu name on Mac OS X:
Example:

```
dim mb as new MenubarMBS
dim m as MenuMBS = mb.item(1) // 1 is in my tests the app menu
if m<>Nil then
m.MenuTitle = "Hello World"
end if
```

Notes: This code is for Carbon only.

20.0.69 How to change the name in the menubar of my app on Mac OS X?

Plugin Version: all, Platform: macOS.

Answer:

You mean it screws up if the file name of the bundle itself is different than the name of the executable file in the MacOS folder within the bundle? If so, you should find something like this within your Info.plist file (or the 'plst' resource that the RB IDE builds for you):

<key>CFBundleExecutable</key>
<string>Executable file name here</string>

Just make sure that file name matches.

However, if your question involves how you can change the name of the app that appears in the menu and the dock, that's different. You can make this name different from the file name by changing the CFBundleName key:

<key>CFBundleName</key>
<string>Name for menu here</string>

Note that if you use my free AppBundler program, this second part is taken care of for you – just fill in a custom name in the right field. You can find AppBundler (from Thomas Reed) at http://www.bitjuggler.com/products/appbundler/ .

20.0.70 How to check if a folder/directory has subfolders?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this to check all items in a folder:
Example:

```
Function HasSubFolder(folder as FolderItem) As Boolean
dim c as Integer = folder.Count

for i as Integer = 1 to c
dim item as FolderItem = folder.TrueItem(i)

if item<>Nil and item.Directory then
Return true
end if
```

```
next

End Function
```

Notes: We use trueitem() here to avoid resolving alias/link files.
Also we check for nil as we may not have permission to see all items.
And if one is a directory, we return without checking the rest.

20.0.71 How to check if Macbook runs on battery or AC power?

Plugin Version: all, Platform: macOS.

Answer: Please use our IOPowerSourcesMBS class like this:
Example:

```
Function PowerSourceState() as Integer
dim p as new IOPowerSourcesMBS

// check all power sources
dim u as Integer = p.Count-1
for i as Integer = 0 to u
dim d as CFDictionaryMBS = p.Item(i)
if d<>nil then
// check if they have a power source state key:
dim o as CFObjectMBS = d.Value(NewCFStringMBS("Power Source State"))
if o isa CFStringMBS then
dim s as string = CFStringMBS(o).str

'MsgBox s

if s = "AC Power" then
Return 1
elseif s = "Battery Power" then
Return 2
end if
end if
end if
next
Return 0 // unknown
End Function
```

Notes: If you want to check the CFDictionaryMBS content, simply use a line like "dim x as dictionary = d.dictionary" and check the contents in the debugger.

20.0.72 How to check if Microsoft Outlook is installed?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you need Outlook for Scripting, you should simply check registry for the required Outlook.Application class:
Example:

```
Function OutlookInstalled() As Boolean
#if TargetWin32 then

try
dim r as new RegistryItem("HKEY_CLASSES_ROOT\Outlook.Application\CLSID", false)

Return true

catch r as RegistryAccessErrorException
// not installed
Return false

end try

#else

// Windows only, so false on other platforms
Return false

#endif

End Function
```

20.0.73 How to check on Mac OS which country or language is currently selected?

Plugin Version: all, Platform: macOS.

Answer: The code below returns a country value.
Example:

```
dim result as Integer

IF TargetMacOS THEN
```

```
CONST smScriptLang = 28
CONST smSystemScript = -1

DECLARE FUNCTION GetScriptManagerVariable LIB "Carbon" ( selector as Integer) as Integer
DECLARE FUNCTION GetScriptVariable LIB "Carbon" ( script as Integer, selector as Integer) as Integer

result=GetScriptVariable(smSystemScript, smScriptLang)

END IF
```

Notes: Returns values like:

For more values, check "Script.h" in the frameworks.

20.0.74 How to code sign my app with plugins?

Plugin Version: all, Platform: macOS.

Answer: When you try to code sign the application with plugin dylibs on Mac OS X, you may see error message that there is actually a signature included.
Notes: Please use the -f command line parameter with codesign utility to overwrite our MBS signature.
We sign our plugins for MacOS, iOS and Windows to make sure they have not been modified.

In terminal, you do like this:

cd <Path to folder of app>

xattr -cr <Appname>.app
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app/Contents/Frameworks/*.dylib
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app/Contents/Frameworks/*.framework
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app

Please use the name of your certificate (See keychain), the name of your app and the path to the app folder.
If you have helper apps you need to sign them first.
You can use a build step to automatically sign your app on build.

20.0.75 How to collapse a window?

Plugin Version: all, Platform: macOS.

Answer: Use this function (Mac only):
Example:

```
Sub CollapseRBwindow(w as window, CollapseStatus as boolean)
dim state, err as Integer
dim wh as MemoryBlock

Declare Function CollapseWindow Lib "Carbon" (window as Integer,collapse as Integer) as Integer

IF CollapseStatus THEN
state = 1
ELSE
state = 0
END IF

err = CollapseWindow(w.MacWindowPtr, state)

End Sub
```

Notes: Also the MBS Plugin has a window.collapsedmbs property you can set.
For Windows the MBS Plugin has a window.isiconicmbs property.

20.0.76 How to compare two pictures?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
Function ComparePictures(p as picture,q as picture) as Integer
dim r,u as RGBSurface
dim x,y,n,m,h,w as Integer
dim w1,w2,h1,h2,d1,d2 as Integer
dim c1,c2 as color

h1=p.Height
h2=q.Height
w1=p.Width
w2=q.Width
d1=p.Depth
d2=q.Depth

if d1<>d2 then
Return 1
elseif w1<>w2 then
```

```
return 2
elseif h1<>h2 then
Return 3
else
r=p.RGBSurface
u=q.RGBSurface

if r=nil or u=nil then
Return -1
else
h=h1-1
w=w1-1
m=min(w,h)

for n=0 to m
c1=r.Pixel(n,n)
c2=u.Pixel(n,n)
if c1<>c2 then
Return 4
end if
next

for y=0 to h
for x=0 to w
c1=r.Pixel(x,y)
c2=u.Pixel(x,y)
if c1<>c2 then
Return 5
end if
next
next

// 0 for equal
// -1 for error (no RGBsurface)
// 1 for different depth
// 2 for different width
// 3 for different height
// 4 for different pixels (fast test)
// 5 for different pixels (slow test)
end if
end if

Exception
Return -1
End Function
```

Notes: Remember that this only works on bitmap pictures, so the picture.BitmapMBS function may be useful.

20.0.77 How to compile PHP library?

Plugin Version: all, Platform: macOS.

Answer: You have to download the source code and compile a static version of the library.
Notes: This instructions were written based on PHP 5.2.6 on Mac OS X:

- Best take a new Mac with current Xcode version installed.
- Download the source code archive. e.g. "php-5.2.6.tar.bz2"
- Expand that archive on your harddisc.
- Open terminal window
- change directory to the php directory. e.g. "cd /php-5.2.6"
- execute this two lines to define the supported CPU types and the minimum Mac OS X version:
- export CFLAGS="-arch ppc -arch i386 -mmacosx-version-min=10.3"
- export CXXFLAGS="-arch ppc -arch i386 -mmacosx-version-min=10.3"
- the command "./configure help" does show the configure options.
- use configure with a line like this:
- ./configure –enable-embed –with-curl -enable-ftp –enable-zip –enable-sockets –enable-static –enable-soap –with-zlib –with-bz2 –enable-exif –enable-bcmath –enable-calendar
- start the compilation with "make all"
- other option is to use "make install" which first does the same as "make all" and than does some installation scripts.
- you may get an error about a duplicate symbole _yytext. Search the file "zend_ini_scanner.c", search a line with "char *yytext;" and change it to "extern char *yytext;".
- On the end you get a lot of error messages, but you have a working library (named libphp5.so) file in the invisible ".libs" folder inside your php source folder.

Possible problems and solutions:

- If the path to your files has spaces, you can get into trouble. e.g. "/RB Plugins/PHP" is bad as files will be searched sometimes in "/RB".

- If you have in /usr/local/lib libraries which conflict with the default libraries, you can get into trouble.
- If you installed some open source tools which compiled their own libraries, you can get into conflicts.
- if you have to reconfigure or after a problem, you may need to use ”make clean” before you start ”make all” again.

Feel free to install additional libraries and add more packages to the configure line.

20.0.78 How to convert a BrowserType to a String with WebSession.Browser?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetBrowserName(s as WebSession.BrowserType) As string
Select case s
case WebSession.BrowserType.Android
Return ”Andriod”
case WebSession.BrowserType.Blackberry
Return ”Blackberry”
case WebSession.BrowserType.Chrome
Return ”Chrome”
case WebSession.BrowserType.ChromeOS
Return ”ChromeOS”
case WebSession.BrowserType.Firefox
Return ”Firefox”
case WebSession.BrowserType.InternetExplorer
Return ”InternetExplorer”
case WebSession.BrowserType.Opera
Return ”Opera”
case WebSession.BrowserType.Safari
Return ”Safari”
case WebSession.BrowserType.SafariMobile
Return ”SafariMobile”
case WebSession.BrowserType.Unknown
Return ”Unknown”
else
Return ”Unkown: ”+str(integer(s))
end Select

End Function
```

20.0.79 How to convert a EngineType to a String with WebSession.Engine?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetRenderingEngineName(s as WebSession.EngineType) As string
Select case s
case WebSession.EngineType.Gecko
Return "Gecko"
case WebSession.EngineType.Presto
Return "Presto"
case WebSession.EngineType.Trident
Return "Trident"
case WebSession.EngineType.Unknown
Return "Unknown"
case WebSession.EngineType.WebKit
Return "WebKit"
else
Return "Unkown: "+str(integer(s))
end Select

End Function
```

20.0.80 How to convert a PlatformType to a String with WebSession.Platform?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetPlatformName(s as WebSession.PlatformType) As string
Select case s
case WebSession.PlatformType.Blackberry
Return "Blackberry"
case WebSession.PlatformType.iPad
Return "iPad"
case WebSession.PlatformType.iPhone
Return "iPhone"
case WebSession.PlatformType.iPodTouch
Return "iPodTouch"
case WebSession.PlatformType.Linux
Return "Linux"
case WebSession.PlatformType.Macintosh
Return "Macintosh"
```

```
case WebSession.PlatformType.PS3
Return "PS3"
case WebSession.PlatformType.Unknown
Return "Unknown"
case WebSession.PlatformType.WebOS
Return "WebOS"
case WebSession.PlatformType.Wii
Return "Wii"
case WebSession.PlatformType.Windows
Return "Windows"
else
Return "Unkown: "+str(integer(s))
end Select

End Function
```

20.0.81 How to convert a text to iso-8859-1 using the TextEncoder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you althrough it's not perfect.
You need to set lc to the current color you use.
Example:

```
dim outstring as string
dim theMac, thePC as textencoding
dim Mac2PC as textconverter

theMac = getTextEncoding(0) // MacRoman
thePC = getTextEncoding(&h0201) // ISOLatin1

Mac2PC = getTextConverter(theMac, thePC)
// if you wanted to do the opposite just create a converter
// PC2Mac = getTextConverter(thePC, theMac)

outstring = Mac2PC.convert("Bj√rn, this text should be converted")
Mac2PC.clear
```

Notes:

You have to call Mac2PC.clear after every conversion to reset the encoding engine.
See also newer TextConverterMBS class.

20.0.82 How to convert ChartTime back to Xojo date?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have this example code:
Example:

```
Function ChartTimeToDate(ChartTime as Double) As date
static diff as Double = 0.0

if diff = 0.0 then
dim d2 as Double = CDBaseChartMBS.chartTime(2015, 1, 1)
dim da as new date(2015, 1, 1)
dim ts as Double = da.TotalSeconds

diff = ts - d2
end if

dim d as new date
d.TotalSeconds = diff + ChartTime

Return d
End Function
```

Notes: As you see we calculate the difference in base date from Date and ChartTime and later use difference to convert.

20.0.83 How to convert line endings in text files?

Plugin Version: all, Platform: macOS.

Answer: You can simply read file with TextInputStream and write with new line endings using TextOutputStream class.
Example:

```
dim inputfile as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim outputfile as FolderItem = SpecialFolder.Desktop.Child("output.txt")
dim it as TextInputStream = TextInputStream.Open(inputfile)
dim ot as TextOutputStream = TextOutputStream.Create(outputfile)

ot.Delimiter = EndOfLine.Windows // new line ending
while not it.EOF
ot.WriteLine it.ReadLine
wend
```

Notes: TextInputStream will read any input line endings and with delimiter property in TextOutputStream you can easily define your new delimiter.

20.0.84 How to convert picture to string and back?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use this plugin functions:
Notes: JPEG:

JPEGStringToPictureMBS(buf as string) as picture
JPEGStringToPictureMBS(buf as string,allowdamaged as Boolean) as picture
PictureToJPEGStringMBS(pic as picture,quality as Integer) as string

PNG:

PictureToPNGStringMBS(pic as picture, gamma as single) as string
PictureToPNGStringMBS(pic as picture, mask as picture, gamma as single) as string
PictureToPNGStringMBS(pic as picture, gamma as single, Interlace as Boolean, FilterType as Integer) as string
PictureToPNGStringMBS(pic as picture, mask as picture, gamma as single, Interlace as Boolean, FilterType as Integer) as string
PNGStringToPictureMBS(data as string, gamma as single) as picture
PNGStringToPNGPictureMBS(data as string, gamma as single) as PNGpictureMBS

Tiff:

TIFFStringToPictureMBS(data as string) as picture
TIFFStringToTiffPictureMBS(data as string) as TiffPictureMBS

BMP:

BMPStringtoPictureMBS(data as string) as picture
Picture.BMPDataMBS(ResolutionValueDPI as Integer=72) as string

GIF:

GifStringToGifMBS(data as string) as GIFMBS
GifStringToPictureMBS(data as string) as Picture

20.0.85 How to copy an array?

Plugin Version: all, Platform: macOS.

Answer: You can use a function like this to copy an array:
Example:

```
Function CopyArray(a() as Double) as Double()
dim r() as Double
for each v as Double in a
r.Append v
next
Return r
End Function
```

Notes: If needed make several copies of this method with different data types, not just double.
For a deep copy of an array of objects, you need to change code to also make a copy of those objects.

20.0.86 How to copy an dictionary?

Plugin Version: all, Platform: macOS.

Answer: You can use a function like this to copy a dictionary:
Example:

```
Function CopyDictionary(d as Dictionary) As Dictionary
dim r as new Dictionary
for each key as Variant in d.keys
r.Value(key) = d.Value(key)
next
Return r
End Function
```

Notes: If needed make several copies of this method with different data types, not just double.
For a deep copy of an dictionary of objects, you need to change code to also make a copy of those objects.

20.0.87 How to copy parts of a movie to another one?

Plugin Version: all, Platforms: macOS, Windows.

Answer: The code below copies ten seconds of the snowman movie to the dummy movie starting at the 5th second.

Example:

```
dim f as FolderItem
dim md as EditableMovie
dim ms as EditableMovie

f=SpecialFolder.Desktop.Child("Our First Snowman.mov")
ms=f.OpenEditableMovie

ms.SelectionStartMBS=5
ms.SelectionLengthMBS=10

f=SpecialFolder.Desktop.Child("dummy.mov")
md=f.CreateMovie

msgbox str(md.AddMovieSelectionMBS(ms))
```

Notes: If result is not 0, the method fails.

20.0.88 How to create a birthday like calendar event?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
// start a connection to the calendar database
dim s as new CalCalendarStoreMBS

// needed for the error details
dim e as NSErrorMBS

dim r as CalRecurrenceRuleMBS = CalRecurrenceRuleMBS.initYearlyRecurrence(1, nil) // repeat every year without end

dim a as new CalAlarmMBS // add alarm
a.action = a.CalAlarmActionDisplay
a.relativeTrigger = -3600*24 // 24 Hours before

// create a new calendar
dim c as new CalEventMBS

dim d as new date(2011, 04, 20) // the date

dim calendars() as CalCalendarMBS = s.calendars
```

```
// set properties
c.Title="Test Birthday"
c.startDate=d
c.recurrenceRule = r
c.calendar=calendars(0) // add to first calendar
c.addAlarm(a)
c.endDate = d
c.isAllDay = true

// save event
call s.saveEvent(c,s.CalSpanAllEvents, e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox "New event was created."
end if
```

Notes: This adds an event to iCal for the given date with alarm to remember you and repeats it every year.

20.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

20.0.90 How to create a Mac picture clip file?

Plugin Version: all, Platform: Windows.

Answer: You can use code like this one.
Example:

```
dim f As FolderItem
dim p As Picture

f=SpecialFolder.Desktop.Child("Test.pictClipping")
if f=nil then Return

p=new Picture(300,200,32) 'Make a sample picture
p.Graphics.ForeColor=RGB(0,255,255)
p.Graphics.FillOval 0,0,99,99
```

```
p.Graphics.ForeColor=RGB(255,0,0)
p.Graphics.DrawOval 0,0,99,99

dim r As ResourceFork 'ResourceFork is needed for a clip file

// Please define a file type Any
r=f.CreateResourceFork("Any")

// get PICT data using plugin function
dim pictdata as string = p.PicHandleDataMBS
r.AddResource(pictdata,"PICT",256,"Picture")

dim m as new MemoryBlock(8)

m.LittleEndian = false
m.Int16Value(0) = 0
m.Int16Value(2) = 0
m.Int16Value(4) = p.Width
m.Int16Value(6) = p.Height

r.AddResource(m,"RECT",256,"")

'Values taken from a sample file and irrelevant to the problem
dim data as string = DecodeBase64("AQAAAAAAAAAAAAAAAAACAFRDRVIAAAABAAAAAAAAAABUQ0lQAAAAA
r.AddResource(data,"drag",128,"") 'ditto
r.Close
```

Notes: In general Apple has deprecated this, but a few application still support clippings.

20.0.91 How to create a PDF file in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Check our DynaPDF plugin and the examples.
Notes: An alternative can be to use the CoreGraphics and Cocoa functions on Mac OS X.
For Windows, we can only suggest our DynaPDF plugin.

20.0.92 How to create EmailAttachment for PDF Data in memory?

Plugin Version: all, Platform: macOS.

Answer: You can use code like the one below:
Example:

```
Function EmailAttachmentFromPDFData(PDFData as string, filename as string) As EmailAttachment
dim a as new EmailAttachment

a.data = EncodeBase64(PDFData, 76)
a.ContentEncoding = "base64"
a.MIMEType = "application/pdf"
a.MacType = "PDF "
a.MacCreator = "prvw"
a.Name = filename

Return a
End Function
```

Notes: Compared to sample code from Xojo documentation, we set the mime type correct for PDF.
The MacType/MacCreator codes are deprecated, but you can still include them for older Mac email clients.
"prvw" is the creator code for Apple's preview app.

20.0.93 How to create PDF for image files?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use DynaPDF like this:
Example:

```
Function CreatePrintPDF(jpgFiles() as folderitem, pdfFile as FolderItem, PageWidth as Integer, PageHeight
as Integer) As Boolean
// have files?
If pdfFile = Nil Then Return False
If jpgFiles = Nil Then Return False

If jpgFiles.Ubound <0 Then Return False

// new DynaPDF
Dim pdf As New MyDynapdfMBS

// page width/height in MilliMeter
Dim pdfWidth as Integer = PageWidth * 72 / 25.4
Dim pdfHeight as Integer = PageHeight * 72 / 25.4

// put your license here
Call pdf.SetLicenseKey "Starter"

// create pdf
Call pdf.CreateNewPDF pdfFile
```

```
// set a couple of options
Call pdf.SetPageCoords(MyDynaPDFMBS.kpcTopDown)
Call pdf.SetResolution(300)
Call pdf.SetUseTransparency(False)
Call pdf.SetSaveNewImageFormat(False)
Call pdf.SetGStateFlags(MyDynaPDFMBS.kgfUseImageColorSpace, False)
Call pdf.SetJPEGQuality(100)

// set page size
Call pdf.SetBBox(MyDynaPDFMBS.kpbMediaBox, 0, 0, pdfWidth, pdfHeight)
Call pdf.SetPageWidth(pdfWidth)
Call pdf.SetPageHeight(pdfHeight)

// append pages with one image per page
For i as Integer = 0 To jpgFiles.Ubound
Call pdf.Append
Call pdf.InsertImageEx(0, 0, pdfWidth, pdfHeight, jpgFiles(i), 1)
Call pdf.EndPage
Next

// close
Call pdf.CloseFile

Return True
End Function
```

Notes: This is to join image files in paper size to a new PDF.
e.g. scans in A4 into an A4 PDF.

20.0.94 How to CURL Options translate to Plugin Calls?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below a few tips on how to translate command line CURL calls to plugin calls.
Notes: curl -vX PUT http://localhost:5984/appserials/78569238475/DocumentRegister.docx?rev=3-25634563456 –data-binary @DocumentRegister.docx -H ”Content-Type: application/msword”

- The option -v means verbose. You can use OptionVerbose and listen for messages in the DebugMessage event.
- The option -X PUT means we want to do a HTTP PUT Request. So set OptionPut to true. Also you will want to set OptionUpload to true as you upload data.
- We have the URL which you put into OptionURL property.

- The –data-binary option tells CURL to pass the given data. With the @ before the data, it is intrepreted as a file name, so the data is read from the given file. You'll need to open this file and pass data with the Read event as needed. (See CURLS ftp file upload example project)
- The last option -H specifies an additional header for the upload. Pas this additional header with the SetOptionHTTPHeader method.

curl -X PUT http://127.0.0.1:5984/appserials/f2f4e540bf8bb60f61cfcd4328001c59 -d ' { "type":"Product","description":"Application Serial","acronym":"AppSerial","dateAdded":"2011-03-21 14:57:36" } '

- Option -X PUT like above.
- Pass the URL again in OptionURL
- This time data is passed in command line for CURL. You'd put this data in the quotes into a string and make it available in the Read event. (See CURLS ftp upload example project)

20.0.95 How to delete file with ftp and curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can set post/pre quotes to have ftp commands executed before or after the download/upload.
Example:

```
dim d as CURLMBS // your curl object

// delete file
dim ws() As String
ws.Append "DELE Temp.txt"

d.SetOptionPostQuote(ws)
```

Notes: Use SetOptionPostQuote, SetOptionPreQuote or SetOptionQuote.
The ftp commands you pass here are native ftp commands and not the commands you use with ftp applications. To delete use DELE and the file path.

20.0.96 How to detect display resolution changed?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac OS X simply listen for display changed notifications.
Notes: Use the "Distribution Notification Center.rbp" example project as a base and use it to listen to notifications with the name "O3DeviceChanged".

20.0.97 How to detect retina?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use Window.BackingScaleFactorMBS to query the factor.
Example:

msgbox str(window1.BackingScaleFactorMBS)

20.0.98 How to disable force quit?

Plugin Version: all, Platform: macOS.

Answer:

Please visit this website and get the control panel for Mac OS 9 there:
http://www3.sk.sympatico.ca/tinyjohn/DFQ.html

For Mac OS X use the MBS Plugin with the SetSystemUIModeMBS method.
Notes:

Please use presentationOptions in NSApplicationMBS for Cocoa applications.

20.0.99 How to disable the error dialogs from Internet Explorer on javascript errors?

Plugin Version: all, Platform: Windows.

Answer: You can use this code in the htmlviewer open event:
Example:

```
if targetwin32 then
htmlviewer1._ole.Content.value("Silent") = True
end if
```

Notes: This disables the error dialogs from Internet Explorer.

20.0.100 How to display a PDF file in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac OS X you can use CoreGraphics or PDFKit to display a PDF.
Notes: An alternative can be to load the PDF into a htmlviewer so the PDF plugin can display it.
On Windows you may need to use the Acrobat ActiveX control from Adobe or launch Acrobat Reader.

20.0.101 How to do a lottery in RB?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this function:
Example:

```
Sub Lotto(max as Integer,count as Integer,z() as Integer)
// Lotto count numbers of max put into the array z beginning at index 0
dim n(0) as Integer ' all the numbers
dim m as Integer ' the highest field in the current array
dim i,a,b,d as Integer ' working variables

'fill the array with the numbers
m=max-1
redim n(m)

for i=0 to m
n(i)=i+1
next

' unsort them by exchanging random ones
m=max*10
for i=1 to m
a=rnd*max
b=rnd*max

d=n(a)
n(a)=n(b)
n(b)=d
next

' get the first count to the dest array
m=count-1
redim z(m)
for i=0 to m
z(i)=n(i)
next

'sort the result
z.sort
End Sub
```

```
Sub Open()
// Test it

dim za(0) as Integer ' the array of the numbers

lotto 49,6,za ' 6 of 49 in Germany


' and display them
staticText1.text=str(za(0))+chr(13)+str(za(1))+chr(13)+str(za(2))+chr(13)+str(za(3))+chr(13)+str(za(4))+chr(13)+str(za
End Sub
```

20.0.102 How to do an asycron DNS lookup?

Plugin Version: all, Platform: Windows.

Answer: use CFHostMBS class (Mac OS X only).
Notes: Xojo internal functions and plugin DNS functions are sycronized.
You can use DNSLookupThreadMBS class for doing them asyncron.

20.0.103 How to draw a dushed pattern line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
// call like this: DrawDushedPatternLine g,0,0,width,height,10

Sub DrawDushedPatternLine(g as graphics,x1 as Integer,y1 as Integer,x2 as Integer,y2 as Integer, partlen
as Integer)
dim x,y,ox,oy as Double
dim dx,dy as Double
dim w,h,d as Double
dim b as Boolean

w=x2-x1
h=y2-y1

d=sqrt(w*w+h*h)

dx=w/d*partlen
dy=h/d*partlen
```

```
b=true
x=x1
while (x<x2) and (y<y2)
ox=x
oy=y

x=x+dx
y=y+dy

if b then
g.DrawLine ox,oy,x,y
end if

b=not b
wend


End Sub
```

Notes: It would be possible to add this to the plugin, but I think it's better if you do it in plain Xojo code, so it even works on Windows.

20.0.104 How to draw a nice antialiased line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you althrough it's not perfect.
You need to set lc to the current color you use.
Example:

```
Sub drawLine(xs as Integer, ys as Integer, xe as Integer, ye as Integer, face as RGBSurface, lineColor as color)
dim intX, intY, count, n, xDiff, yDiff as Integer
dim v, v1, floatX, floatY, xx, yy, xStep, yStep as Double
dim c as color

const st=1.0

xDiff=xe-xs
yDiff=ye-ys
count=max(abs(xDiff), abs(yDiff))
xStep=xDiff/count
yStep=yDiff/count
```

```
xx=xs
yy=ys
for n=1 to count
intX=xx
intY=yy
floatX=xx-intX
floatY=yy-intY

v=(1-floatX)*(1-floatY)*st
v1=1-v
c=face.pixel(intX, intY)
face.pixel(intX, intY)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=floatX*(1-floatY)*st
v1=1-v
c=face.pixel(intX+1, intY)
face.pixel(intX+1, intY)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=(1-floatX)*floatY*st
v1=1-v
c=face.pixel(intX, intY+1)
face.pixel(intX, intY+1)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=floatX*floatY*st
v1=1-v
c=face.pixel(intX+1, intY+1)
face.pixel(intX+1, intY+1)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)

xx=xx+xStep
yy=yy+yStep
next

End Sub
```

Notes:

PS: st should be 1 and face should be a RGBSurface or a Graphics object.

20.0.105 How to dump java class interface?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In terminal you can use ”javap -s <classname>” to display the class with the method names and parameters.
Notes: For example show ResultSet class: javap -s java.sql.ResultSet

20.0.106 How to duplicate a picture with mask or alpha channel?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this function:
Example:

```
Function Duplicate(extends p as Picture) As Picture
#if RBVersion >= 2011.04 then
if p.HasAlphaChannel then

// create nw picture and copy content:
dim q as new Picture(p.Width, p.Height)
q.Graphics.DrawPicture p,0,0

Return q

end if
#endif

// create new picture
dim q as new Picture(p.Width, p.Height, 32)

// get mask
dim oldMask as Picture = p.mask(false)
if oldMask = nil then
// no mask, so simple copy
q.Graphics.DrawPicture p,0,0
Return q
end if

// remove mask
p.mask = nil

// copy picture and mask
q.Graphics.DrawPicture p, 0, 0
q.mask.Graphics.DrawPicture oldMask,0,0

// restore mask
p.mask = oldmask

Return q
End Function
```

Notes: Simply copy it to a module and call it like this: q = p.duplicate.
The code above works with old Xojo versions because of the #if even if your RS version does not support alpha channel pictures. This way it's future proof.

20.0.107 How to enable assistive devices?

Plugin Version: all, Platform: macOS.

Answer: You can use AppleScript code like below:
Notes: tell application "System Events"
activate

set UI elements enabled to true

return UI elements enabled
end tell

You can run this with AppleScriptMBS class.

20.0.108 How to encrypt a file with Blowfish?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
dim fi as FolderItem = SpecialFolder.Desktop.Child("test.xojo_binary_project")
dim fo as FolderItem = SpecialFolder.Desktop.Child("test.encrypted")

// read input
dim bi as BinaryStream = BinaryStream.Open(fi)
dim si as string = bi.Read(bi.Length)
bi.Close

// encrypt
dim so as string = BlowfishMBS.Encrypt("MyKey",si)

// write output
dim bo as BinaryStream = BinaryStream.Create(fo)
bo.Write so
bo.Close
```

Notes: Of course you can decrypt same way, just use Decrypt function and of course swap files.

20.0.109 How to extract text from HTML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use both RemoveHTMLTagsMBS and DecodingFromHTMLMBS like this:
Example:

```
dim html as string = "<p><B>Gr&uuml;&szlig;e</B></P>"
dim htmltext as string = RemoveHTMLTagsMBS(html)
dim text as string = DecodingFromHTMLMBS(htmltext)

MsgBox text // shows: Gr√º√üe
```

Notes: You can use it together with RemoveHTMLTagsMBS to remove html tags. What you get will be the text without tags.
DecodingFromHTMLMBS turns HTML escapes back to unicode characters. Like ä to √§.

20.0.110 How to find empty folders in a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this code:
Example:

```
dim folder as folderitem // your folder

dim c as Integer = folder.count
for i as Integer = 1 to c
dim item as folderitem = folder.trueitem(i)
if item = nil then
// ignore
elseif item.directory then
// folder
if item.count = 0 then
// found empty folder
end if
end if
next
```

20.0.111 How to find iTunes on a Mac OS X machine fast?

Plugin Version: all, Platform: macOS.

Answer: You can try Launch Services.
Example:

```
dim f as FolderItem

f=LaunchServicesFindApplicationForInfoMBS("hook","com.apple.iTunes","iTunes.app")

MsgBox f.NativePath
```

20.0.112 How to find network interface for a socket by it's name?

Plugin Version: all, Platform: macOS.

Answer: You can use our plugin to build a lookup table.
Example:

```
Function FindNetworkInterface(name as string) As NetworkInterface
name = name.trim

if name.len = 0 then Return nil

// search by IP/MAC
dim u as Integer = System.NetworkInterfaceCount-1
for i as Integer = 0 to u
dim n as NetworkInterface = System.GetNetworkInterface(i)
if n.IPAddress = name or n.MACAddress = name then
Return n
end if
next

// use MBS Plugin to build a mapping
dim interfaces() as NetworkInterfaceMBS = NetworkInterfaceMBS.AllInterfaces
dim map as new Dictionary

for each n as NetworkInterfaceMBS in interfaces
dim IPv4s() as string = n.IPv4s
dim IPv6s() as string = n.IPv6s

for each IPv4 as string in IPv4s
map.Value(IPv4) = n.Name
next
for each IPv6 as string in IPv6s
map.Value(IPv6) = n.Name
next
if n.MAC<>"" then
map.Value(n.MAC) = n.Name
```

```
end if
next

// now search interfaces by name, IPv4 or IPv6
for i as Integer = 0 to u
dim n as NetworkInterface = System.GetNetworkInterface(i)
if map.Lookup(n.IPAddress, "") = name then
Return n
end if

if map.Lookup(n.MACAddress, "") = name then
Return n
end if
next

End Function
```

Notes: The code above uses a lookup table build using NetworkInterfaceMBS class to find the network interface by name.

20.0.113 How to find version of Microsoft Word?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
// find Word
dim f as FolderItem = LaunchServicesFindApplicationForInfoMBS("","com.microsoft.Word","")

// open bundle
dim c as new NSBundleMBS(f)

// read info
dim d as Dictionary = c.infoDictionary

// show version
MsgBox d.Lookup("CFBundleVersion","")
```

Notes: Older versions of Word can be found with creator code "MSWD".

20.0.114 How to fix CURL error 60/53 on connecting to server?

Plugin Version: all, Platform: macOS.

Answer: You probably connect with SSL and you have no valid certificate.
Example:

```
dim d as new CURLSMBS

// Disable SSL verification
d.OptionSSLVerifyHost = 0 // don't verify server
d.OptionSSLVerifyPeer = 0 // don't proofs certificate is authentic

// With SSL Verification:
dim cacert as FolderItem = Getfolderitem("cacert.pem")
d.OptionCAInfo = cacert.NativePath
d.OptionSSLVerifyHost = 2 // verify server
d.OptionSSLVerifyPeer = 1 // proofs certificate is authentic
```

Notes: You can either use the code above to disable the SSL verification and have no security.
Or you use the cacert file and enable the verification. Than you only get a connection if the server has a valid certificate.

see also:
http://curl.haxx.se/ca/

20.0.115 How to format double with n digits?

Plugin Version: all, Platform: macOS.

Answer: You can use the FormatMBS function for this.
Example:

```
dim d as Double = 123.4567890
listbox1.AddRow FormatMBS("%f", d)
listbox1.AddRow FormatMBS("%e", d)
listbox1.AddRow FormatMBS("%g", d)

listbox1.AddRow FormatMBS("%5.5f", d)
listbox1.AddRow FormatMBS("%5.5e", d)
listbox1.AddRow FormatMBS("%5.5g", d)

d = 0.000000123456
listbox1.AddRow FormatMBS("%f", d)
listbox1.AddRow FormatMBS("%e", d)
```

```
listbox1.AddRow FormatMBS("%g", d)

listbox1.AddRow FormatMBS("%5.5f", d)
listbox1.AddRow FormatMBS("%5.5e", d)
listbox1.AddRow FormatMBS("%5.5g", d)
```

Notes: see FormatMBS for details.
In general %f is normal style, %e is scientific and %g is whichever gives best result for given space.

20.0.116 How to get a time converted to user time zone in a web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebSession.GMTOffset property.
Example:

```
Sub Open()
// current date on server
dim d as new date
dim s as string = d.LongTime

// adjust to client GMT offset
d.GMTOffset = d.GMTOffset + Session.GMTOffset

dim t as string = D.LongTime

MsgBox s+EndOfLine+t
End Sub
```

20.0.117 How to get an handle to the frontmost window on Windows?

Plugin Version: all, Platform: Windows.

Answer: This function returns a handle for the frontmost window:
Example:

```
Function GetForegroundWindowHandle() as Integer
#if targetwin32 then
declare function GetForegroundWindow Lib "user32.dll" as Integer
Return GetForegroundWindow()
#endif
End Function
```

20.0.118 How to get CFAbsoluteTime from date?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Use code like this:
Example:

```
dim d as new date
dim t as CFTimeZoneMBS = SystemCFTimeZoneMBS
dim g as new CFGregorianDateMBS
g.Day = d.Day
g.Month = d.Month
g.Year = d.Year
g.Minute = d.Minute
g.Hour = d.Hour
g.Second = d.Second

dim at as CFAbsoluteTimeMBS = g.AbsoluteTime(t)
dim x as Double = at.Value

MsgBox str(x)
```

Notes: As you see we need a timezone and put the date values in a gregorian date record.
Now we can query absolute time for the given timezone.

20.0.119 How to get client IP address on web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebSession.RemoteAddress property.
Example:

```
Sub Open()
Title = Session.RemoteAddress
End Sub
```

20.0.120 How to get fonts to load in charts on Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use the SetFontSearchPath method in the CDBaseChartMBS class to specify where your fonts are.
Example:

```
if TargetLinux then
CDBaseChartMBS.SetFontSearchPath "/usr/share/fonts/truetype;/usr/share/fonts/truetype/msttcorefonts"
else
// on Mac and Windows we use system fonts.
end if

// also you can later switch default fonts:

dim Chart as CDBaseChartMBS // your chart

#If TargetARM And TargetLinux Then
// use specific fonts on Linux on Raspberry Pi
Call Chart.setDefaultFonts("/usr/share/fonts/truetype/piboto/PibotoLt-Regular.ttf","/usr/share/fonts/truetype/piboto/Pi
#EndIf
```

Notes: On macOS, iOS and Windows, the fonts are loaded from the system's font folder.

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

20.0.121 How to get fonts to load in DynaPDF on Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use the AddFontSearchPath method in the DynaPDFMBS class to specify where your fonts are.
Example:

```
dim d as new DynaPDFMBS
if TargetLinux then
call d.AddFontSearchPath "/usr/share/fonts/truetype", true
else
// on Mac and Windows we use system fonts.
end if
```

Notes: On Mac OS X and Windows, the fonts are loaded from the system's font folder.

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

20.0.122 How to get GMT time and back?

Plugin Version: all, Platform: macOS.

Answer: You can use the date class and the GMTOffset property.
Example:

```
// now
dim d as new date

// now in GMT
dim e as new date
e.GMTOffset = 0

// show
MsgBox str(d.TotalSeconds,"0.0")+" "+str(e.TotalSeconds, "0.0")

dim GMTTimeStamp as Double = e.TotalSeconds

// restore
dim f as new date

// add GMT offset here
f.TotalSeconds = GMTTimeStamp + f.GMTOffset*3600
// because here it's removed
f.GMTOffset = f.GMTOffset

MsgBox d.ShortTime+" ("+str(d.GMTOffset)+") "+str(d.TotalSeconds,"0.0")+EndOfLine+_
e.ShortTime+" ("+str(e.GMTOffset)+") "+str(e.TotalSeconds,"0.0")+EndOfLine+_
f.ShortTime+" ("+str(f.GMTOffset)+") "+str(f.TotalSeconds,"0.0")
```

Notes: It's sometimes a bit tricky with the date class as setting one property often changes the others.

20.0.123 How to get good crash reports?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Check this website from the webkit website:
Notes: http://webkit.org/quality/crashlogs.html

20.0.124 How to get list of all threads?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the runtime module like in this function:
Example:

```
Function Threads() As Thread()
#pragma DisableBackgroundTasks
dim t() as Thread

Dim o as Runtime.ObjectIterator=Runtime.IterateObjects
While o.MoveNext
if o.Current isa Thread then
t.Append thread(o.current)
end if
Wend

Return t
End Function
```

Notes: This returns an array of all thread objects currently in memory.
The pragma is important here as it avoids thread switches which may cause a thread to be created or deleted.

20.0.125 How to get parameters from webpage URL in Xojo Web Edition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the Webpage.ParametersReceived event.
Example:

```
Sub ParametersReceived(Variables As Dictionary)
for each key as Variant in Variables.keys
MsgBox key+" ->"+Variables.Value(key)
next
End Sub
```

Notes: The text encodings of this strings is not defined in Xojo 2010r5. Please use DefineEncoding.

20.0.126 How to get the color for disabled textcolor?

Plugin Version: all, Platform: macOS.

Answer: Ask the appearance manager:
Example:

```
Function GetThemeTextColor(inColor as Integer, inDepth as Integer, inColorDev as Boolean) As Color
declare function GetThemeTextColor lib "Carbon" (inColor as Integer, inDepth as Integer, inColorDev as Boolean, outColor as Ptr) as Integer

dim i as Integer
dim col as MemoryBlock

col = newMemoryBlock(6)

i = GetThemeTextColor(inColor, inDepth, inColorDev, col)

return RGB(col.UShort(0)\256, col.UShort(2)\256, col.UShort(4)\256)
End Function
```

Notes: The color for this is:

```
const kThemeTextColorDialogInactive = 2.

c = GetThemeTextColor(kThemeTextColorDialogInactive, Screen(0).Depth, true)
```

For Mac OS X you should use "CarbonLib" instead of "AppearanceLib" ...

20.0.127 How to get the current free stack space?

Plugin Version: all, Platform: macOS.

Answer: You can something like the code below:
Example:

```
Sub ShowStackSize()
dim threadid as Integer
dim size as Integer

declare function GetCurrentThread lib "Carbon" (byref threadid as Integer) as short
declare function ThreadCurrentStackSpace lib "Carbon" (threadid as Integer, byref size as Integer) as short

if GetCurrentThread(threadid)=0 then
if 0=ThreadCurrentStackSpace(threadid,size) then
MsgBox str(size)
end if
end if
```

```
End Sub
```

Notes: For Mac OS 9, use "ThreadLib" instead of "CarbonLib". You can use #if if you like for that.

20.0.128 How to get the current timezone?

Plugin Version: all, Platforms: macOS, Windows.

Answer:

You can use the TimeZoneMBS class or the CFTimeZoneMBS class.
Or code like below:
Example:

```
Function GMTOffsetInMinutes() as Integer
// Returns the offset of the current time to GMT in minutes.
// supports Mac OS and Windows, but not Linux yet (let me know if
// you have code for that, please)
//
// Note that the offset is not always an even multiple of 60, but
// there are also half hour offsets, even one 5:45h offset

// This version by Thomas Tempelmann (rb@tempel.org) on 25 Nov 2005
// with a fix that should also make it work with future Intel Mac targets.
//
// Using code from various authors found on the RB NUG mailing list

dim result, bias, dayLightbias as Integer
dim info as memoryBlock
dim offset as Integer

#if targetMacOS then

Declare Sub ReadLocation lib "Carbon" (location As ptr)

info = NewMemoryBlock(12)
ReadLocation info
if false then
// bad, because it does not work on Intel Macs:
'offset = info.short(9) * 256 + info.byte(11)
else
offset = BitwiseAnd (info.long(8), &hFFFFFF)
end

offset = info.short(9) * 256 + info.byte(11)
```

```
offset = offset \60
return offset

#endif

#if targetWin32 then

Declare Function GetTimeZoneInformation Lib "Kernel32" ( tzInfoPointer as Ptr ) as Integer
// returns one of
// TIME_ZONE_ID_UNKNOWN 0
// – Note: e.g. New Delhi (GMT+5:30) and Newfoundland (-3:30) return this value 0
// TIME_ZONE_ID_STANDARD 1
// TIME_ZONE_ID_DAYLIGHT 2

info = new MemoryBlock(172)
result = GetTimeZoneInformation(info)

bias = info.Long(0)
// note: the original code I found in the NUG archives used Long(84) and switched to Long(0)
// only for result=1 and result=2, but my tests found that Long(0) is also the right value for result=0

if result = 2 then
daylightBias = info.long(168)
end if
offset = - (bias + dayLightbias)
return offset

#endif

End Function
```

20.0.129 How to get the current window title?

Plugin Version: all, Platform: macOS.

Answer: The code below returns the current window title for the frontmost window on Mac OS X if Accessibilty services are
Example:

```
Function CurrentWindowTitle() As string
// your application needs permissions for accessibility to make this work!

dim SystemWideElement,FocusedApplicationElement,FocusedWindowElement as AXUIElementMBS
dim FocusedApplication,FocusedWindow,Title as AXValueMBS
dim s as String
dim cs as CFStringMBS
```

```
SystemWideElement=AccessibilityMBS.SystemWideAXUIElement
if SystemWideElement<>nil then
FocusedApplication=SystemWideElement.AttributeValue(AccessibilityMBS.kAXFocusedApplicationAttribute)
if FocusedApplication.Type=AccessibilityMBS.kAXUIElementMBSTypeID then
FocusedApplicationElement=new AXUIElementMBS
FocusedApplicationElement.Handle=FocusedApplication.Handle
FocusedApplicationElement.RetainObject

FocusedWindow=FocusedApplicationElement.AttributeValue(AccessibilityMBS.kAXFocusedWindowAttribute)

if FocusedWindow<>nil and AccessibilityMBS.kAXUIElementMBSTypeID=FocusedWindow.Type then

FocusedWindowElement=new AXUIElementMBS
FocusedWindowElement.Handle=FocusedWindow.Handle
FocusedWindowElement.RetainObject

Title=FocusedWindowElement.AttributeValue(AccessibilityMBS.kAXTitleAttribute)
if Title<>nil and Title.Type=kCFStringMBSTypeID then
cs=new CFStringMBS
cs.handle=Title.Handle
cs.RetainObject
Return cs.str
end if
end if
end if
end if
End Function
```

20.0.130 How to get the cursor blink interval time?

Plugin Version: all, Platform: macOS.

Answer: On Mac OS you can use GetCaretTime from the toolbox.
Example:

```
declare function GetCaretTime lib "Carbon" () as Integer

MsgBox str(GetCaretTime())+" ticks"
```

Notes: 60 ticks make one second.

20.0.131 How to get the list of the current selected files in the Finder?

Plugin Version: all, Platform: macOS.

Answer:

Use the AppleScript like this one:

```
tell application "finder"
return selection
end tell
```

Which translates into this AppleEvent:

```
Process("Finder").SendAE "core,getd,'—-':obj { form:prop, want:type(prop), seld:type(sele), from:'null'() }
"
```

and as Xojo code it looks like this:
Example:

```
dim ae as appleevent
dim o1 as appleeventObjectSpecifier
dim f as folderItem
dim aList as appleeventdescList
dim i as Integer
dim dateiname as string

// setup the AppleEvent
o1=getpropertyObjectDescriptor( nil, "sele")
ae= newappleEvent("core", "getd", "MACS")
ae.objectSpecifierParam("—-")=o1

// send it
if ae.send then
// got the list
alist=ae.replyDescList

// now show the list of filename into an editfield:

for i=1 to alist.count
f=alist.folderItemItem(i)

dateiname=f.name
// editfield1 with property "mulitline=true"!
editfield1.text=editfield1.text + dateiname + chr(13)
next
```

```
end if
```

20.0.132 How to get the Mac OS system version?

Plugin Version: all, Platform: macOS.

Answer: The following code queries the value and displays the version number:
Example:

```
dim first as Integer
dim second as Integer
dim third as Integer
dim l as Integer

if System.Gestalt("sysv",l) then

Third=Bitwiseand(l,15)
second=Bitwiseand(l\16,15)
first=Bitwiseand(l\256,15)+10*Bitwiseand(l\256\16,15)
end if

if First>=10 then
msgbox "Mac OS X "+str(First)+"."+str(Second)+"."+str(third)
else
msgbox "Mac OS "+str(First)+"."+str(Second)+"."+str(third)
end if
```

20.0.133 How to get the Mac OS Version using System.Gestalt?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
Dim s As String
Dim b As Boolean
Dim i, resp as Integer

// Systemversion
b = System.Gestalt("sysv", resp)
If b then
s = Hex(resp)
```

```
For i =Len(s)-1 DownTo 1
s=Left(s,i)+"."+Mid(s,i+1)
Next
MsgBox "Systemversion: Mac OS " + s
end if
```

Notes: The MBS Plugin has a SystemInformationMBS.OSVersionString function for this.

20.0.134 How to get the screensize excluding the task bar?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Notes: Use the Screen class with the available* properties.

20.0.135 How to get the size of the frontmost window on Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Notes: Make yourself a class for the WindowRect with four properties:

```
Bottom as Integer
Left as Integer
Right as Integer
Top as Integer
```

Add the following method to your class:

```
Sub GetWindowRect(windowhandle as Integer)
dim err as Integer
dim mem as memoryBlock
#if targetwin32 then
Declare Function GetWindowRect Lib "user32.dll" (hwnd as Integer, ipRect As Ptr) as Integer

mem = newmemoryBlock(16)
err = GetWindowRect(windowhandle, mem)
Left = mem.long(0)
Top = mem.Long(4)
Right = mem.Long(8)
Bottom = mem.Long(12)
```

```
#endif
End Sub
```

Good to use for the MDI Master Window!

20.0.136 How to get the source code of a HTMLViewer?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
// for Windows:

msgbox HTMLViewer1.IEHTMLTextMBS

// for MacOS with WebKit 2.x:

msgbox HTMLViewer1.WKWebViewMBS.HTMLText
```

20.0.137 How to get Xojo apps running Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You need to install some requuire packages.
Notes: You need CUPS as well as GTK packages. On 64 bit systems also the ia32-libs package.

Please note that you need a x86 compatible Linux. So no PPC, Power, ARM or other CPUs.

20.0.138 How to handle really huge images with GraphicsMagick or ImageMagick?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sometimes it may be better to use an extra application to process images.
Notes: A typical 32 bit app made with Xojo can use around 1.8 GB on Windows and 3 GB on Mac OS X. Some images may be huge, so that processing them causes several copies of the image to be in memory. With a 500 MB image in memory, doing a scale or rotation may require a temp image. So with source, temp and dest images with each 500 MB plus your normal app memory usage, you may hit the limit of Windows with 1.8 GB.

In that case it may be worth running a tool like gm in the shell class. gm is the command line version of GraphicsMagick. There you can run the 64 bit version which is not limited in memory like your own application. Also you can monitor progress and keep your app responsive.

20.0.139 How to handle tab key for editable cells in listbox?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this function:
Example:

```
Function HandleTabInList(list as listbox, row as Integer, column as Integer, key as String) As Boolean
// Handle tab character in Listbox.CellKeyDown event

Select case asc(key)
case 9
if Keyboard.AsyncShiftKey then
// back

// look for column left
for i as Integer = column-1 downto 0
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next

// not found, so look in row before
row = row - 1
if row >= 0 then
for i as Integer = list.ColumnCount-1 downto 0
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
end if
else
// forward

// look for column right
for i as Integer = column+1 to list.ColumnCount-1
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
```

```
// not found, so look in row below
row = row + 1
if row <list.ListCount then
for i as Integer = 0 to list.ColumnCount-1
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
end if
end if
end Select
End Function
```

Notes: You call it from CellKeyDown event like this:

```
EventHandler Function CellKeyDown(row as Integer, column as Integer, key as String) As Boolean
if HandleTabInList(me, row, column, key) then Return true
End EventHandler
```

As you see in the code, we handle tab and shift + tab for moving back and forward. Also we wrap to previous/next row if needed. Feel free to extend this to wrap from last to first row or create a new row for editing.

20.0.140 How to hard link MapKit framework?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Our MapKit classes weak link the framework. If you need hard linking it for the App Store, you can add this method to a class:
Example:

```
Sub ReferenceMapKit()
// just put this in window or app class

#if TargetMachO and Target64Bit then
Declare sub testing Lib "MapKit" Selector "test" (id as ptr)
testing(nil)
#endif

End Sub
```

Notes: No need to call the method.
Just having it in a window or app, will cause the compiler to hard link the framework.

20.0.141 How to have a PDF downloaded to the user in a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use a WebHTMLViewer control and load the PDF file with the PDF plugin from the browser.
Example:

```
dim CurrentFile as WebFile // a property of the WebPage

// define the PDF file
CurrentFile = new WebFile
CurrentFile.Filename = "test.pdf"
CurrentFile.MIMEType = "application/pdf"
CurrentFile.Data = "some pdf data" // MyDynaPDF.GetBuffer
CurrentFile.ForceDownload = true

// start the download
showurl(CurrentFile.url)
```

Notes: See our Create PDF example for the Xojo Web Edition.

20.0.142 How to hide all applications except mine?

Platform: macOS.

Answer: The code below will on Mac OS hide all applications except your one:
Example:

```
dim p as new ProcessMBS

p.GetFirstProcess
do
if not p.FrontProcess then
p.Visible=false
end if
loop until not p.GetNextProcess
```

20.0.143 How to hide script errors in HTMLViewer on Windows?

Plugin Version: all, Platform: Windows.

Answer: Set Internet Explorer to silent mode with code like this:
Example:

```
htmlviewer1._ole.Content.value("Silent") = True
```

Notes: Simply put this code in the open event of your htmlviewer control (using me instead of htmlviewer1).

20.0.144 How to hide the grid/background/border in ChartDirector?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you want to hide something in a chart, simply assign the kTransparent constant as color.

20.0.145 How to hide the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub HideCursor Lib "Carbon" () Inline68K("A852")

HideCursor
```

Notes: The MBS Plugin has this function and supports it on Windows, too.

20.0.146 How to insert image to NSTextView or TextArea?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to insert file:
Example:

```
// insert a file to textview

Public Sub InsertFile(textview as NSTextViewMBS, f as FolderItem)
// read to file
```

```
dim b as BinaryStream = BinaryStream.Open(f)
dim s as string = b.Read(b.Length)

// build wrapper
dim fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
dim fileAttachment as new NSTextAttachmentMBS(fileWrapper)
dim attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttach-
ment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS method.

20.0.147 How to jump to an anchor in a htmlviewer?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You can use javascript to change the current window's location.
Example:

```
// load website
htmlviewer1.LoadURL "http://www.monkeybreadsoftware.net/addressbook-abpersonmbs.shtml"

// later jump to anchor named "16":

if TargetWin32 then
call HTMLViewer1.IERunJavaScriptMBS "window.location = ""#16"""
end if
```

20.0.148 How to keep a movieplayer unclickable?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: To keep the user away from clicking on a playing Movie you can just drop a Canvas in front of the Movieplayer and take the clicks there.
Example:

```
Function Canvas1.MouseDown(X as Integer, Y as Integer) as boolean
return true // take it and do nothing
End Function
```

20.0.149 How to keep my web app from using 100% CPU time?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Linux and MacOS you can use renice command in the terminal. On Windows use the task manager to reduce priority.
Notes: If you launch your app with nohup on Linux or Mac OS X like this from the terminal or a script:

nohup /webapps/MyApp/MyApp &

you can simply have a second line saying this:

renice 20 $!

which tells the system to lower priority to lowest value for the latest background process.

20.0.150 How to kill a process by name?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can kill a process (or application) by name if you loop over all the processes and kill the one you need.
Example:

```
dim p as new ProcessMBS
p.GetfirstProcess ' get first
do
if p.name = "TextEdit" then
call p.KillProcess
Return
end if
loop until not p.GetNextProcess
```

Notes: You may want to check the result of killProcess function. Not every user is allowed to kill every application.

20.0.151 How to know how many CPUs are present?

Plugin Version: all, Platform: macOS.

Answer: Try this function:
Example:

```
Function GetCPUCount() as Integer
Declare Function MPProcessors Lib "Carbon" () as Integer

Return MPProcessors()
End Function
```

Notes: Your app will than need that library to launch on Classic. To avoid this the MBS plugin checks if this library is available and return 1 if it's not available.

20.0.152 How to know the calling function?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac you can use a helper function like this this code:
Example:

```
Public Function CallingFunction() as string
// Query name of calling function of a function

#Pragma BreakOnExceptions false

try

// raise a dummy exception
dim r as new NilObjectException
raise r

catch x as NilObjectException

// get stack
dim stack() as string = x.Stack

// pick function name and return
dim name as string = stack(2)
Return name

end try
End Function
```

Notes: You need to include function names in your application.

20.0.153 How to launch an app using it's creator code?

Plugin Version: all, Platform: macOS.

Answer: Send an AppleEvent "oapp" with the creator code to the Finder ("MACS"):
Example:

```
Dim a as AppleEvent
dim creator as string

creator = "MSIE" ' here the Internet Explorer

a = NewAppleEvent("aevt", "odoc", "MACS")
a.Timeout = -1

a.ObjectSpecifierParam("—-") = GetUniqueIDObjectDescriptor("appf", nil, creator)

if not a.send then
msgBox "An error has occured"
else

end if
```

20.0.154 How to launch disc utility?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use this code:
Example:

```
dim f as FolderItem = LaunchServicesFindApplicationForInfoMBS("","com.apple.DiskUtility","")

if f<>Nil then
f.Launch
end if
```

Notes: This works even if people renamed the disc utility or moved it to another folder.

20.0.155 How to make a lot of changes to a REAL SQL Database faster?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You may try to embed your changes to the database between two transaction calls.
Example:

```
dim db as Database // some database

db.SQLExecute "BEGIN TRANSACTION"
// Do some Stuff
db.SQLExecute "END TRANSACTION"
```

Notes: This can increase speed by some factors.

20.0.156 How to make a NSImage object for my retina enabled app?

Plugin Version: all, Platform: macOS.

Answer: You can use code like this:
Example:

```
Function NewRetinaImage(pic as Picture, mask as Picture = nil) As NSImageMBS
// first make a NSImageMBS from it
dim n as new NSImageMBS(pic, mask)

// now set to half the size, so we have 2x pixels for the image
n.size = new NSSizeMBS(n.width/2, n.height/2)

// and return
Return n
End Function
```

Notes: The thing to do is to have 2x the pixels, but assign a size to the image which gives it the right size in points.
You can pass the NSImageMBS from here to NSMenuItemMBS. For Retina displays, the full resolution is used. For others it will be reduced.

20.0.157 How to make a window borderless on Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this declares:
Example:

```
// Sets window to borderless popup type, and sets its initial dimensions.
// Call this method, then Win32SetBorderlessPos, and then RB's Show
// method. Use RB Frame type 7 (Global Floating Window).

Const SWP_NOMOVE = &H2
Const SWP_FRAMECHANGED = &H20
Const HWND_TOPMOST = -1
Const GWL_STYLE = -16
Const WS_POPUPWINDOW = &H80880000

Dim styleFlags as Integer

#If TargetWin32 Then

Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (hwnd as Integer, nIndex as Integer, dwNewLong as Integer) as Integer
Declare Function SetWindowPos Lib "user32" (hwnd as Integer, hWndInstertAfter as Integer, x as Integer, y as Integer, cx as Integer, cy as Integer, flags as Integer) as Integer

styleFlags = SetWindowLong( w.WinHWND, GWL_STYLE, WS_POPUPWINDOW )
styleFlags = BitwiseOr( SWP_FRAMECHANGED, SWP_NOMOVE )
styleFlags = SetWindowPos( w.WinHWND, HWND_TOPMOST, 0, 0, wd, ht, styleFlags )

#EndIf
```

20.0.158 How to make an alias using AppleEvents?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
Sub MakeAlias(folder as folderitem, target as folderitem, aliasname as string)
dim ev as AppleEvent
dim myResult as boolean
dim properties as AppleEventRecord

ev = NewAppleEvent("core","crel","MACS")
ev.MacTypeParam("kocl") = "alis"
ev.FolderItemParam("to ") = target
ev.FolderItemParam("insh") = folder

properties=new AppleEventRecord
```

```
properties.StringParam("pnam")=aliasname

ev.RecordParam("prdt")=properties

myResult = ev.send
// true on success, false on error
End Sub
```

Notes: Call it like this:
MakeAlias SpecialFolder.Desktop, SpecialFolder.Desktop.Child("Gif Copy.rb"), "test.rb alias"

Seems to not work on Mac OS X 10.6

20.0.159 How to make AppleScripts much faster?

Plugin Version: all, Platform: macOS.

Answer: use "ignoring application responses" like in this example:
Notes: on run { fn,fpx,fpy }
ignoring application responses
tell app "Finder" to set the position of folder fn to fpx,fpy
end ignoring
end run

20.0.160 How to make double clicks on a canvas?

Plugin Version: all, Platform: macOS.

Answer:

Update: Newer Xojo versions support DoubleClick event, so you don't need this code.

Here's my tip from the tips list on how to add a double-click event to the Canvas control. The technique could easily be used for a window or any Rectcontrol:

Because of its built-in drawing methods, the Canvas control is often used to create custom interface controls. But while the Canvas control has event handlers for most mouse events, it doesn't have an event handler for DoubleClick events. Fortunately, you can add a double-click event handler to a Canvas control easily. Basically, you're going to create a new class based on Canvas and add a double-click event to that. You can then use the new class anytime you need a Canvas with a double-click event.

To create a new Canvas class with a DoubleClick event handler, do this:

1. Add a new class to your project.
2. Set the Super property of the new class to "Canvas".
3. Change the name of this new class to "DoubleClickCanvas".

A double-click occurs when two clicks occur within the users double-click time (set in the Mouse control panel on both Macintosh and Windows) and within five pixels of each other. So, you'll need a few properties to store when and where the last click occurred.

4. Add a new property with the following declaration and mark it as private: lastClickTicks as Integer
5. Add a new property with the following declaration and mark it as private: lastClickX as Integer
6. Add a new property with the following declaration and mark it as private: lastClickY as Integer

Since the Canvas control doesn't have a DoubleClick event, you will need to add one.

7. Add a new event to your class by choosing New Event from the Edit menu and enter "DoubleClick" as the event name.

Double-clicks occur on MouseUp. In order for the mouseUp event to fire, you must return True in the MouseDown event.

8. In the MouseDown event, add the following code:
Return True

In the MouseUp event, you will need to determine what the users double-click time is. This value is represented on both the Mac and Windows in ticks. A tick is 1/60th of a second. Since there isn't a built-in function for this, you'll need to make a toolbox call. The mouseUp event code below makes the appropriate toolbox call for both Macintosh and Windows. It then compares the time of the users last click to the time of the current click and compares the location of the users last click to the location of the current click.

9. Add the following code to the MouseUp event:

```
dim doubleClickTime, currentClickTicks as Integer

#if targetMacOS then
Declare Function GetDblTime Lib "Carbon" () as Integer
doubleClickTime = GetDblTime()
#endif

#if targetWin32 then
Declare Function GetDoubleClickTime Lib "User32.DLL" () as Integer
```

```
doubleClickTime = GetDoubleClickTime()/60 // convert to ticks from milliseconds
#endif

currentClickTicks = ticks
//if the two clicks happened close enough together in time
if (currentClickTicks - lastClickTicks) <= doubleClickTime then
//if the two clicks occured close enough together in space
if abs(X - lastClickX) <= 5 and abs(Y - LastClickY) <= 5 then
DoubleClick //a double click has occured so call the event
end if
end if
lastClickTicks = currentClickTicks
lastClickX = X
lastClickY = Y
```

10. Now to test out your new DoubleClickCanvas, drag the class from the Project window to a window in your project to create an instance of it.

11. Double-click on the canvas you just added to your window to open the Code Editor. Notice that the canvas has a DoubleClick event handler. In this event handler, add the following code:

```
BEEP
```

20.0.161 How to make my Mac not sleeping?

Plugin Version: all, Platform: macOS.

Answer: Just inform the Mac OS about some system activity with code like this:
Example:

```
Sub UpdateSystemActivity()

#if TargetCarbon
declare function myUpdateSystemActivity lib "Carbon" alias "UpdateSystemActivity" (activity as Integer)
as short

const OverallAct = 0 // Delays idle sleep by small amount */
const UsrActivity = 1 // Delays idle sleep and dimming by timeout time */
const NetActivity = 2 // Delays idle sleep and power cycling by small amount */
const HDActivity = 3 // Delays hard drive spindown and idle sleep by small amount */
const IdleActivity = 4 // Delays idle sleep by timeout time */

dim e as Integer

e=myUpdateSystemActivity(UsrActivity)
```

```
// you may react on an error if e is not 0 after the call.

#endif
End Sub
```

Notes: You may use another constant if you prefer some different behavior.
Call it maybe every second.

20.0.162 How to make my own registration code scheme?

Plugin Version: all, Platform: Windows.

Answer: There are excellent articles about how to make a registratin code scheme, but you can also simply use our RegistrationEngineMBS class.
Notes: If you need a license text, why not use the one from Xojo as a starting point?

20.0.163 How to make small controls on Mac OS X?

Plugin Version: all, Platform: macOS.

Answer: You can try this code on Mac OS X:
Example:

```
'/*
'* Use the control's default drawing variant. This does not apply to
'* Scroll Bars, for which Normal is Large.
'*/
const kControlSizeNormal = 0

'/*
'* Use the control's small drawing variant. Currently supported by
'* the Check Box, Combo Box, Radio Button, Scroll Bar, Slider and Tab
'* controls.
'*/
const kControlSizeSmall = 1

'/*
'* Use the control's small drawing variant. Currently supported by
'* the Indeterminate Progress Bar, Progress Bar and Round Button
'* controls.
'*/
const kControlSizeLarge = 2
```

```
'/*
'* Control drawing variant determined by the control's bounds. This
'* ControlSize is only available with Scroll Bars to support their
'* legacy behavior of drawing differently within different bounds.
'*/
const kControlSizeAuto = &hFFFF

const kControlSizeTag = "size"

declare function SetControlData lib "Carbon" (controlhandle as Integer, part as short, tagname as OSType, size as Integer, data as ptr) as short

dim m as MemoryBlock

m=NewMemoryBlock(2)
m.UShort(0)=kControlSizeSmall

Title=str(SetControlData(CheckBox1.Handle, 0, kControlSizeTag, 2, m))
```

20.0.164 How to mark my Mac app as background only?

Plugin Version: all, Platform: macOS.

Answer: You can run a build script on each build with this code:
Example:

```
Dim App As String = CurrentBuildLocation + "/" + CurrentBuildAppName + ".app"
Call DoShellCommand("/usr/bin/defaults write " + App + "/Contents/Info ""NSUIElement"" YES")
```

Notes: This will set the NSUIElement flag to YES.

20.0.165 How to move a file or folder to trash?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like below:
Example:

```
Function MoveToTrash(f as FolderItem) As Boolean
#if TargetMacOS then
dim r as FolderItem
dim e as Integer = MacFileOperationMBS.MoveObjectToTrashSync(f, r, MacFileOperationMBS.kFSFileOperationDefaultOptions)
```

```
if e = 0 then
Return true // Ok
end if

#elseif TargetWin32 then
dim w as new WindowsFileCopyMBS

dim flags as Integer = w.FileOperationAllowUndo + w.FileOperationNoErrorUI + w.FileOperationSilent
+ w.FileOperationNoConfirmation
if w.FileOperationDelete(f, flags) then
Return true // OK
end if

flags = w.FileOperationNoErrorUI + w.FileOperationSilent + w.FileOperationNoConfirmation
if w.FileOperationDelete(f, flags) then
Return true // OK
end if
#else
// Target not supported
break
Return false
#endif
End Function
```

Notes: If you want to move a file to trash, you could use f.movefileto f.trashfolder, but that will overwrite existing files in the trash. You can use our MacFileOperationMBS class to move a file on Mac to the trash. And it uses the same code as the Finder, so files are renamed when the same name is already in use in the trash:

On Windows we use WindowsFileCopyMBS class.
Requires Mac OS X 10.5.

20.0.166 How to move an application to the front using the creator code?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code ttxt) to the frontmost application:
Example:

```
dim a as appleevent

a=newappleEvent("misc","actv","ttxt")
```

```
if a.send then
end if
```

Notes: (Code is Mac only)

20.0.167 How to move file with ftp and curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can set post/pre quotes to have ftp commands executed before or after the download/upload.
Example:

```
dim d as CURLMBS // your curl object

// rename/move file
dim ws() As String
ws.Append "RNFR Temp.txt"
ws.append "RNTO MyFile.txt"

d.SetOptionPostQuote(ws)
```

Notes: Use SetOptionPostQuote, SetOptionPreQuote or SetOptionQuote.
The ftp commands you pass here are native ftp commands and not the commands you use with ftp applications. So rename is two commands. First RNFR to tell where to rename from and second RNTO with the new file name. To delete use DELE and the file path.

20.0.168 How to normalize string on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like below:
Example:

```
Function Normalize(t as string) As string
const kCFStringNormalizationFormD = 0 // Canonical Decomposition
const kCFStringNormalizationFormKD = 1 // Compatibility Decomposition
const kCFStringNormalizationFormC = 2 // Canonical Decomposition followed by Canonical Composition
const kCFStringNormalizationFormKC = 3 // Compatibility Decomposition followed by Canonical Composition

dim s as CFStringMBS = NewCFStringMBS(t)
dim m as CFMutableStringMBS = s.Normalize(kCFStringNormalizationFormD)
```

```
Return m.str
End Function
```

Notes: This uses Apple's CFString functions to normalize unicode variants.

20.0.169 How to obscure the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub ObscureCursor Lib "Carbon" ()

ObscureCursor
```

Notes: The MBS Plugin has this function, but it's not supported for Windows.

20.0.170 How to open icon file on Mac?

Plugin Version: all, Platform: macOS.

Answer: Use the NSImageMBS class like this:
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.ico")
dim n as new NSImageMBS(f)

window1.Backdrop = n.CopyPictureWithMask
```

20.0.171 How to open PDF in acrobat reader?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim pdf as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
```

```
// open PDF in Acrobat Reader on Mac:

// find app
dim bundleID as string = "com.adobe.Reader"
dim app as FolderItem = LaunchServicesFindApplicationForInfoMBS("", bundleID, "")

if app<>nil then

// launch app with parameters

dim docs() as FolderItem
docs.Append pdf

dim param as new LaunchServicesLaunchParameterMBS
param.Defaults = true
param.Application = app

dim x as FolderItem = LaunchServicesOpenXMBS(docs, param)

// on failure, simply launch it
if x = nil then
pdf.Launch(true)
end if

else
pdf.Launch(true)
end if
```

Notes: On Windows, simply use pdf.launch or WindowsShellExecuteMBS.

20.0.172 How to open printer preferences on Mac?

Plugin Version: all, Platform: macOS.

Answer: You can use our OpenMacOSXPreferencesPaneMBS function like this:
Example:

```
dim e as Integer = OpenMacOSXPreferencesPaneMBS("PrintAndFax")
if 0 = e then
MsgBox "OK"
elseif e = -43 then
MsgBox "File not found."
else
MsgBox "Error: "+str(e)
end if
```

20.0.173 How to open special characters panel on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have functions for that in Cocoa and Carbon.
Example:

```
dim a as new NSApplicationMBS
a.orderFrontCharacterPalette
```

Notes: For Cocoa, you can use orderFrontCharacterPalette method in NSApplicationMBS class.

Or simply for Carbon and Cocoa the ShowCharacterPaletteMBS method.

20.0.174 How to optimize picture loading in Web Edition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebPicture class.
Notes: Take your picture and create a WebPicture object. Store this WebPicture in a property of the WebPage, Session or app (as global as possible). On the first time you use this picture on an user session, the browser will load it. Second time you use it, the browser will most likely pick it from the cache.
Having pictures in App or some module reuses the same picture for all sessions which reduces memory footprint.
This does not work well with pictures you change very often or use only for one webpage on one user.

If you like to see an example, check our Map example.

20.0.175 How to parse XML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
dim s as string = "<test><test /></test>"

try
```

```
dim x as new XmlDocument(s)
MsgBox "OK"
catch xe as XmlException
MsgBox "invalid XML"
end try
```

Notes: If you got an exception, you have a parse error.

20.0.176 How to play audio in a web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the HTML5 audio tag and control it with javscript.
Notes: This is just another example app I made today. It plays a christmas song. The audio file is provided by the application to the server, so no external web server is needed and this application can run stand alone. To compile and run you need Xojo 2010r5.

In the open event we search the audio files and open them as binarystreams. We create the two webfile objects. Those webfiles are part of the app class, so we have them globally. There we set the data with the content of our streams. We also define file names and mime types. They are needed so browser know what we have here:

```
audioFileM4V = new WebFile
audioFileM4V.Data = bM.Read(BM.Length)
audioFileM4V.Filename = "music.m4a"
audioFileM4V.MIMEType = "audio/m4a"
```

```
audioFileOGG = new WebFile
audioFileOGG.Data = bO.Read(BO.Length)
audioFileOGG.Filename = "music.ogg"
audioFileOGG.MIMEType = "audio/ogg"
```

Next in the open event of the webpage we have a PageSource control. The location is set to be before content. In the open event we define the html code for this. First we pick the URLs for the audio files. Than we build the html to use the audio tag. As you see, we give it an ID for later use and have it preload automatically. If you add an autoplay tag, you can have the audio play right away. Inside the audio tag we have two sources so we provide audio for both Firefox (OGG) and Safari (MPEG4). Finally we have a text to display if HTML5 audio tag is not supported.

You can set the source in the EditSource event:

dim urlo as string = app.audioFileOGG.URL
dim urlm as string = app.audioFileM4V.URL
me.Source = ”<audio id=””mymusic”” preload=””auto””><source src=””””+urlo+”””” type=””audio/ogg”” /><source src=””””+urlm+”””” type=””audio/mpeg”” />Your browser does not support the audio element.</audio>”

Next in the Play button we execute code to play the audio. This is a short javascript code which searches in the html document for the element with the ID ”mymusic” which is the ID of our audio tag above. Once we got the object, we call it’s play method to start playback.

me.ExecuteJavaScript(”document.getElementById(’mymusic’).play();”)

same for pause:

me.ExecuteJavaScript(”document.getElementById(’mymusic’).pause();”)

and finally for changing volume:

me.ExecuteJavaScript(”document.getElementById(’mymusic’).volume=”+str(me.Value/100.0)+”;”)

20.0.177 How to pretty print xml?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the XML Transform method with the right XLS.
Notes: Learn more here:
http://docs.xojo.com/index.php/XMLDocument.Transform

20.0.178 How to print to PDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: This code below shows how to redirect printing to a PDF file on Mac OS X.
Example:

```
// get Xojo printer setup
dim p as new PrinterSetup

// now put it into NSPrintInfo to manipulate
dim n as new NSPrintInfoMBS
n.SetupString = p.SetupString
```

```
// change destination to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
n.SetSaveDestination(f)

// move back
p.SetupString = n.SetupString

// and print as usual
dim g as Graphics = OpenPrinter(p)
g.DrawString "Hello World", 20, 20
```

Notes: And you can use normal graphics class for that.

20.0.179 How to query Spotlight's Last Open Date for a file?

Plugin Version: all, Platform: macOS.

Answer: You can use a MDItemMBS objec to query this value:
Example:

```
Function LastOpenedDate(Extends F As FolderItem, DefaultOtherDates As Boolean = True) As Date
#If TargetMacOS Then
Dim xMDItem as New MDItemMBS(F)
Dim xDate as Variant

If xMDItem <>Nil Then
xDate = xMDItem.GetAttribute(xMDItem.kMDItemLastUsedDate).DateValue
If xDate IsA Date Then Return xDate
Else
If xDate <>Nil Then Break
End If
#EndIf

If DefaultOtherDates Then
If F.ModificationDate <>Nil Then Return F.ModificationDate
If F.CreationDate <>Nil Then Return F.CreationDate
End If
End Function
```

Notes: Thanks for Josh Hoggan for this example code.

20.0.180 How to quit windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetwin32 then
dim i1,i2,r as Integer
declare function ExitWindowsEx lib "user32" (uFlags as Integer, dwReserved as Integer) as Integer
i1 = 2
i2 = 0
r = ExitWindowsEx(i1,i2)
if r<>0 then
' Error()
end if

#endif
```

Notes: uFlags parameters:

```
'4 = EWX_Force
'0 = EWX_Logoff
'2 = EWX_Reboot
'1 = EWX_shutdown, should shut down computer
```

Also check the ExitWindowsMBS method.

20.0.181 How to read a CSV file correctly?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With all the rules for quotes and delimiters, you can simply use the SplitCommaSeparatedValuesMBS method in our plugins like this:
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.csv")
dim t as TextInputStream = f.OpenAsTextFile

while not t.EOF
dim s as string = t.ReadLine(encodings.ASCII)

dim items() as string = SplitCommaSeparatedValuesMBS(s, ";", """")
```

```
List.AddRow ""
dim u as Integer = UBound(items)
for i as Integer = 0 to u
List.Cell(List.LastIndex,i) = items(i)
next

wend
```

Notes: Please make sure you choose the right text encoding.

20.0.182 How to read the command line on windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetwin32 then
dim line as string
Dim mem as MemoryBlock

Declare Function GetCommandLineA Lib "kernel32" () As Ptr

mem=GetCommandLineA()
s=mem.cstring(0)

#endif
```

Notes: Newer Xojo versions have a system.commandline property.

20.0.183 How to render PDF pages with PDF Kit?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
// choose a file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")

// open it as PDF Document
dim sourceFile as New PDFDocumentMBS(f)
```

```
if sourceFile.handle <>0 then // it is a PDF file

// get upper bound of pages
dim c as Integer = sourceFile.pageCount-1

// from first to last page
for n as Integer = 0 to c

// pick that page
dim page as PDFPageMBS = sourceFile.pageAtIndex(n)

// render to image
dim p as NSImageMBS = page.Render

// and convert to RB picture and display
Backdrop = p.CopyPictureWithMask

next

end if
```

Notes: PDFKit works only on Mac OS X.

20.0.184 How to restart a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent("FNDR","rest","MACS")
if not ae.send then
msgBox "The computer couldn't be restarted."
end if
```

20.0.185 How to resume ftp upload with curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: CURL supports that and you simply need to set the right options.

Notes: First of course OptionUpload must be true. Second OptionFTPAppend must be true so the OptionResumeFrom is used. Store there (or in OptionResumeFromLarge) your start value.
Don't forget to implement the read event and return data there as requested.

20.0.186 How to rotate a PDF page with CoreGraphics?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: This code opens a PDF and draws the first page into a new PDF with 90¬∞ rotation.
Example:

```
// Rotate a PDF page

// our files
dim sourcefile as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
dim destfile as FolderItem = SpecialFolder.Desktop.Child("rotated.pdf")

// open PDF
dim pdf as CGPDFDocumentMBS = sourcefile.OpenAsCGPDFDocumentMBS

// query media size of first page
dim r as CGRectMBS = pdf.MediaBox(1)

// create new PDF
dim c as CGContextMBS = destfile.NewCGPDFDocumentMBS(r,"title","Author","Creator")

// create rotated rectangle
dim nr as new CGRectMBS(0,0,r.Height,r.Width)

// create new page
c.BeginPage nr
c.SaveGState

const pi = 3.14159265

// rotate by 90¬∞
c.RotateCTM pi*1.5

// fix origin
c.TranslateCTM -r.width,0

// draw PDF
c.DrawCGPDFDocument pdf,r,1

// cleanup
c.RestoreGState
c.EndPage
```

```
c = nil

// show in PDF viewer
destfile.Launch
```

Notes: This code is Mac only as it needs CoreGraphics.

20.0.187 How to rotate image with CoreImage?

Plugin Version: all, Platform: macOS.

Answer: Use the code like the one below:
Example:

```
// Rotate image with CoreImage

// load image
dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim image as new CIImageMBS(f)

// rotate 45 degree
dim n as new NSAffineTransformMBS
n.rotateByDegrees(45)

dim TransformFilter as new CIFilterAffineTransformMBS
TransformFilter.inputImage = image
TransformFilter.inputTransform = n

// get result
dim resultImage as CIImageMBS = TransformFilter.outputImage

// for saving to file
dim outputImage as NSImageMBS = resultImage.RenderNSImage(false)

f = SpecialFolder.Desktop.Child("output.png")
dim b as BinaryStream = BinaryStream.Create(f, true)
b.Write outputImage.PNGRepresentation

// as Xojo picture object for display
dim pic as Picture = outputImage.CopyPictureWithMask

Backdrop = pic
```

20.0.188 How to run a 32 bit application on a 64 bit Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Install 32 bit compatibility libraries.
Notes: The package is called ia32-libs for ubuntu (and others).
Some applications need to be run on a 32 bit system as they need some hardware related libraries. Like libUSB or libHID for USB devices.

20.0.189 How to save HTMLViewer to PDF with landscape orientation?

Plugin Version: all, Platform: macOS.

Answer: You can use NSPrintInfoMBS to change the options for PrintToPDFFile function.
Example:

```
// make it landscape
dim n as NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
n.orientation = n.NSLandscapeOrientation

// save html to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
call HTMLViewer1.PrintToPDFFileMBS(f,10,30,10,30)
```

Notes: You may want to reset options later.
This code is only for Mac OS X.

20.0.190 How to save RTFD?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to save to RTFD:
Example:

```
// save text as RTFD including image attachments
dim f as FolderItem = GetSaveFolderItem(FileTypes1.ApplicationRtfd, "test.rtfd")

if f = nil then Return

dim a as NSAttributedStringMBS = textView.textStorage
dim w as NSFileWrapperMBS = a.RTFDFileWrapperFromRange(0, a.length, DocumentAttributes)

dim e as NSErrorMBS
if w.writeToFile(f, e) then
```

```
else
MsgBox e.LocalizedDescription
end if
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS method.

20.0.191 How to save RTFD?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: How to load PDF to htmlviewer on desktop?
Example:

```
Public Sub LoadPDFData(viewer as HTMLViewer, PDFData as string)
Dim base64string As String = EncodeBase64(PDFData)

// remove line endings to make it a big line
base64string = ReplaceLineEndings(base64string, "")

// build data URL
// https://en.wikipedia.org/wiki/Data_URI_scheme
Dim dataURL As String = "data:application/pdf;base64," + base64string

// show in webviewer
HTMLViewer1.LoadURL(dataURL)

// may not work everywhere due to URL length limit
// for Web projects, use WebFile instead!
End Sub
```

Notes: This avoids a temporary file, which may also work.
For Web Apps, please use WebFile.

20.0.192 How to scale a picture proportionally with mask?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: For a proportional scaling, we calculate the new picture size relative to the target maximum size.
Example:

```
Function ProportinalScaledWithMask(extends pic as Picture, Width as Integer, Height as Integer) As Picture
// Calculate scale factor

dim faktor as Double = min( Height / Pic.Height, Width / Pic.Width)

// Calculate new size
dim w as Integer = Pic.Width * faktor
dim h as Integer = Pic.Height * faktor

// create new picture
dim NewPic as new Picture(w,h,32)

// check if we have a mask and clear it
dim m as picture = pic.mask(False)
pic.mask = nil

// draw picture in the new size
NewPic.Graphics.DrawPicture Pic, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height

if m <>nil then
// restore mask and scale it
pic.mask = m
NewPic.mask.Graphics.DrawPicture m, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height
end if


// return result
Return NewPic
End Function
```

Notes: This version handles mask. As you see we actually have to remove mask in order to copy the picture part correctly.

20.0.193 How to scale a picture proportionally?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: For a proportional scaling, we calculate the new picture size relative to the target maximum size.
Example:

```
Function ProportionalScaled(extends pic as Picture, Width as Integer, Height as Integer) As Picture
// Calculate scale factor

dim faktor as Double = min( Height / Pic.Height, Width / Pic.Width)
```

```
// Calculate new size
dim w as Integer = Pic.Width * faktor
dim h as Integer = Pic.Height * faktor

// create new picture
dim NewPic as new Picture(w,h,32)

// draw picture in the new size
NewPic.Graphics.DrawPicture Pic, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height

// return result
Return NewPic
End Function
```

Notes: This does not handle mask, but you can scale the mask the same way and assign it to the new picture.
(see other FAQ entry with mask)

20.0.194 How to scale/resize a CIImageMBS?

Plugin Version: all, Platform: Windows.

Answer: Use the CIFilterLanczosScaleTransform filter to scale down a picture to a specific size.
Example:

```
Dim pic As Picture = LogoMBS(500)
Dim image As CIImageMBS = CIImageMBS.imageWithPicture(pic)

Dim filter As New CIFilterLanczosScaleTransformMBS

Const targetWidth = 600.0
Const targetHeight = 400.0

Dim scale As Double = targetHeight / image.Extent.Height
Dim aspect As Double = targetWidth / (image.Extent.Width * scale)

filter.inputImage = image
filter.inputScale = scale
filter.inputAspectRatio = aspect

Dim result As Picture = filter.outputImage.RenderPicture

Backdrop = result
```

Notes: This is same code as our scaleTo convenience method.

20.0.195 How to scale/resize a picture?

Plugin Version: all, Platform: Windows.

Answer: There are several ways to scale or resize a picture. The easiest way may be the ScaleMBS function in the Picture class.
Example:

```
dim Original,Scaled as Picture

Original=LogoMBS(500)
Scaled=Original.ScaleMBS(100,100,true)
```

Notes: The plugin ways:
- GraphicsMagick can scale/resize.
- CoreImage scale filter may result in the fastest and best images on Mac OS X 10.4.
- NSImageMBS can scale, but is Mac OS X only.
- CGImageMBS can scale, but is Mac OS X only.
- CIImageMBS can scale, but is Mac OS X only.
- QuickTime Graphics exporter and importer can be connected to scale. (this was used more often a few years ago)
- ImageMagick can scale very nice and crossplatform. But the ImageMagick libraries are big.
- The picture.ScaleMBS function is self written and results in equal output on Mac, Windows and Linux without any additional libraries installed.
- Picture.ScalingMBS does crossplatform scaling with several modes.

with pure Xojo:
- make a new picture and draw the old one with new size inside.

20.0.196 How to search with regex and use unicode codepoints?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can specify unicode characters in search string with backslash x and digits.
Example:

```
dim r as RegExMbs
dim s as string
dim c as Integer
```

```
s="123 √§√√º ABC 456"

r=new RegExMBS
if r.Compile(".√.") then
c=r.Execute(s,0)
MsgBox str(c)+" "+str(r.Offset(0))+" "+str(r.Offset(1))
// shows: 1 4 10
// 1 for ubound of the offset array
// 4 for 4 bytes before the matched pattern
// 10 for the 10 bytes before the end of the matched pattern
end if

r=new RegExMBS
if r.Compile(".\xF6.") then // finds √ using Unicode codepoint
c=r.Execute(s,0)
MsgBox str(c)+" "+str(r.Offset(0))+" "+str(r.Offset(1))
// shows: 1 4 10
// 1 for ubound of the offset array
// 4 for 4 bytes before the matched pattern
// 10 for the 10 bytes before the end of the matched pattern
end if
```

20.0.197 How to see if a file is invisible for Mac OS X?

Plugin Version: all, Platform: macOS.

Answer: Try this function:
Example:

```
Function Invisible(F As FolderItem) As Boolean
Dim TIS As TextInputStream
Dim S,All As String
Dim I as Integer
dim g as folderitem

If Left(F.Name,1)="." or not f.visible Then
Return True
End If

g=F.Parent.Child(".hidden")
If g.Exists Then
TIS=g.OpenAsTextFile
if tis<>Nil then
All=TIS.ReadAll
For I=1 to CountFields(All,Chr(11))
S=NthField(All, Chr(11), I)
```

```
If S=F.name Then
Return True
End If
Next
end if
End if
End Function
```

20.0.198 How to set cache size for SQLite or REALSQLDatabase?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You use the pragma cache_size command on the database.
Example:

```
// set cache size to 20000 pages which is about 20 MB for default page size
dim db as REALSQLDatabase
db.SQLExecute "PRAGMA cache_size = 20000"
```

Notes: Default cache size is 2000 pages which is not much.
You get best performance if whole database fits in memory.
At least you should try to have a cache big enough so you can do queries in memory.
You only need to call this pragma command once after you opened the database.

20.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:
Example:

```
window1.ModifiedMBS=true
```

20.0.200 How to show a PDF file to the user in a Web Application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use a WebHTMLViewer control and load the
Example:

```
dim CurrentFile as WebFile // a property of the WebPage

// define the PDF file
CurrentFile = new WebFile
CurrentFile.Filename = "test.pdf"
CurrentFile.MIMEType = "application/pdf"
CurrentFile.Data = "some pdf data" // MyDynaPDF.GetBuffer

// load into html viewer
HTMLViewer1.URL = CurrentFile.URL
```

Notes: See our Create PDF example for the Xojo Web Edition.

20.0.201 How to show Keyboard Viewer programmatically?

Platform: macOS.

Answer: Use Xojo or AppleScript to launch the KeyboardViewerServer.app.
Example:

```
dim a as new AppleScriptMBS
dim text as string
dim lines(-1) as string

lines.append "set theApplication to ""KeyboardViewerServer"""
lines.append "set thePath to ""/System/Library/Components/KeyboardViewer.component/Contents/Shared-Support/KeyboardViewerServer.app"""
lines.append ""
lines.append "set POSIXPath to ((POSIX file thePath) as string)"
lines.append "tell application ""System Events"" to set isRunning to 0 <(count (application processes whose name is theApplication))"
lines.append "if isRunning then tell application POSIXPath to quit"
lines.append "delay 0.15"
lines.append ""
lines.append "ignoring application responses"
lines.append " tell application POSIXPath to run"
lines.append "end ignoring"

text=join(lines,EndOfLine.macintosh)

a.Compile text
a.Execute
```

Notes: AppleScript code:

```
set theApplication to "KeyboardViewerServer"
set thePath to "/System/Library/Components/KeyboardViewer.component/Contents/SharedSupport/KeyboardViewerServer.app"

set POSIXPath to ((POSIX file thePath) as string)
tell application "System Events" to set isRunning to 0 <(count (application processes whose name is theApplication))
if isRunning then tell application POSIXPath to quit
delay 0.15

ignoring application responses
tell application POSIXPath to run
end ignoring
```

20.0.202 How to show the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub ShowCursor Lib "Carbon" ()

ShowCursor
```

Notes: The MBS Plugin has this function and supports it on Windows, too.

20.0.203 How to shutdown a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent("FNDR","shut","MACS")
if not ae.send then
msgBox "The computer couldn't be shutdown."
end if
```

Notes: Or toolbox call (Attention: This method will stop the computer immediataly: No document asked to be saved, all applications quitting without knowing).

```
Declare Sub ShutDwnPower Lib "Carbon" ()
ShutDwnPower
```

20.0.204 How to sleep a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent("FNDR","slep","MACS")
if not ae.send then
msgBox "The computer doesn't want to sleep."
end if
```

20.0.205 How to speed up rasterizer for displaying PDFs with DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Here a few speed tips:
Notes:

- Use the DynaPDFRasterizerMBS function instead of our render functions.
- Reuse DynaPDFRasterizerMBS as long as the target picture size doesn't change.
- Import only the PDF pages you want to display.
- Let DynaPDF do zooming, rotating or other effects instead of you change it.

20.0.206 How to use PDFLib in my RB application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The PDFlib plugin was discontinued in favor of our DynaPDF plugin.
Notes: If you need help to move, please contact us.

20.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.
Example:

msgbox "This String contains ""quotes""."

20.0.208 How to use Sybase in Web App?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use our MBS Xojo SQL Plugin to connect to a Sybase Database in your web application.
Notes: If you see db.Connect giving the error message "cs_ctx_alloc ->CS_MEM_ERROR", than some things are not setup right for Sybase.
The Apache process may not have all the SYBASE environment variables being set when the CGI was launched.

Adding these lines to /etc/httpd/conf/httpd.conf stopped the faux memory errors for us:

SetEnv LD_LIBRARY_PATH /opt/sybase/OCS-15_0/lib:/opt/sybase/OCS-15_0/lib3p64:/opt/sybase/OCS-15_0/lib3p:
SetEnv SYBROOT /opt/sybase
SetEnv SYBASE_OCS /opt/sybase
SetEnv SYBASE /opt/sybase

20.0.209 How to use the Application Support folder?

Plugin Version: all, Platform: macOS.

Answer:

I was saving a registration code for an app to the Preferencefolder.
People on the list have suggested that it would be better in
the ApplicationSupportFolder. How do I save the file called CWWPrefs
into that folder using MBS?

I have checked for examples and the docs but can't see how to apply it

//f = SpecialFolder.Preferences.child("CWWPrefs")
f = ApplicationSupportFolderMBS(-32768)

Example:

```
dim folder,file as FolderItem

folder = createApplicationSupportFolderMBS(-32763)

if folder=nil then
// Some very old Mac OS Versions may not support it
// or the plugin may fail for any reason
folder=SpecialFolder.Preferences
end if

file=folder.Child("CWWPrefs")

MsgBox file.NativePath
```

Notes:

You may not be able to write there with a normal user account!

20.0.210 How to use the IOPMCopyScheduledPowerEvents function in Xojo?

Plugin Version: all, Platform: macOS.

Answer: You can use the following code which does this using the SoftDeclareMBS class.
Example:

```
Sub Open()
dim c as CFDateMBS
dim t as CFAbsoluteTimeMBS

// get current date
c=NewCFDateMBS

// in absolute time (seconds since x)
t=c.AbsoluteTime

// add 600 seconds (= 10 Minutes)
t.Value=t.Value+600

// Make a Date from it
c=t.Date

// Schedule the event
// 0 on success
// E00002C1 for missing root rights
```

```
Title=hex(schedulePowerEvent(c, "wake"))

// Just for information, display the scheduled stuff
CFShowMBS CopyScheduledPowerEvents
End Sub

Function CopyScheduledPowerEvents() As cfarrayMBS
dim s as SoftDeclareMBS
dim m as MemoryBlock

s=new SoftDeclareMBS

if s.LoadLibrary("IOKit.framework") then
if s.LoadFunction("IOPMCopyScheduledPowerEvents") then
if s.CallFunction(0,nil) then
Return NewCFArrayMBSHandle(s.Result,true)
else
MsgBox "Failed to Call IOPMCopyScheduledPowerEvents."
end if
else
MsgBox "Failed to load IOPMCopyScheduledPowerEvents."
end if
else
MsgBox "Failed to load IOKit."
end if

Return nil
End Function

Function SchedulePowerEvent(time_to_wake as CFDateMBS, Type as CFStringMBS) as Integer
dim s as SoftDeclareMBS
dim m as MemoryBlock

'/*
'* Types of power event
'* These are potential arguments to IOPMSchedulePowerEvent().
'* These are all potential values of the kIOPMPowerEventTypeKey in the CFDictionaries
'* returned by IOPMCopyScheduledPowerEvents().
'*/
'/*!
'@define kIOPMAutoWake
'@abstract Value for scheduled wake from sleep.
'*/
'#define kIOPMAutoWake "wake"
'
'/*!
'@define kIOPMAutoPowerOn
'@abstract Value for scheduled power on from off state.
```

```
'*/
'#define kIOPMAutoPowerOn "poweron"
'
'/*!
'@define kIOPMAutoWakeOrPowerOn
'@abstract Value for scheduled wake from sleep, or power on. The system will either wake OR
'power on, whichever is necessary.
'*/
'
'#define kIOPMAutoWakeOrPowerOn "wakepoweron"
'/*!
'@define kIOPMAutoSleep
'@abstract Value for scheduled sleep.
'*/
'
'#define kIOPMAutoSleep "sleep"
'/*!
'@define kIOPMAutoShutdown
'@abstract Value for scheduled shutdown.
'*/
'
'#define kIOPMAutoShutdown "shutdown"

s=new SoftDeclareMBS

if s.LoadLibrary("IOKit.framework") then
if s.LoadFunction("IOPMSchedulePowerEvent") then

m=NewMemoryBlock(12)
m.Long(0)=time_to_wake.handle
m.Long(4)=0 // nil
m.Long(8)=type.Handle

if s.CallFunction(3,m) then
Return s.Result
end if
end if
end if

End Function
```

Notes: Requires Mac OS X and to execute root rights.

20.0.211 How to validate a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use this function below which uses a regular expression to verify that the string is a valid UUID/GUID:
Example:

```
Function IsGUID(guid as string) As Boolean
dim r as new RegEx

r.SearchPattern = "^(\{ { 0,1 } ( [ 0-9a-fA-F ] ) { 8 } -( [ 0-9a-fA-F ] ) { 4 } -( [ 0-9a-fA-F ] ) { 4 } -( [ 0-9a-fA-F ] ) { 4 } -( [ 0-9a-fA-F ] ) { 12 } \} { 0,1 } )$ "

Return r.Search(guid)<>nil
End Function
```

Notes: Simply parsing the GUID with CFUUIDMBS does not give the same result as CFUUIDMBS will also take a string like "DDDD".

20.0.212 How to walk a folder hierarchie non recursively?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this one:
Example:

```
Sub Walk(folder as FolderItem)
dim folders() as FolderItem

folders.Append folder

while UBound(folders)>=0

dim currentFolder as FolderItem = folders.pop

dim c as Integer = currentFolder.Count
for i as Integer = 1 to c
dim item as FolderItem = currentFolder.TrueItem(i)

if item = Nil then
// no permission
elseif item.Visible then // only visible

if item.Directory then
folders.Append item
```

```
else
// work with file here
end if

end if

next

wend
End Sub
```

Notes: As you see we go with a long loop which runs until we don't have more folders to process.
We ignore items we can't access due to permission limits.
And we only work visible items.
If you like, check folderitem.isBundleMBS on item to handle packages and applications better on Mac OS X.

20.0.213 I got this error: PropVal, QDPictMBS.Name (property value), Type mismatch error. Expected CGDataProviderMBS, but got Variant, Name:QDPictMBS

Plugin Version: all, Platform: macOS.

Answer: The plugins MacOSX and MacOSXCF belong together. If you use one part, please also install the other part.
Notes: We splitted the plugin because the Xojo IDE on Windows crashed on compilation.

20.0.214 I registered the MBS Plugins in my application, but later the registration dialog is shown.

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: There are two main reasons.
Notes: 1. you may use the plugin before registering them. This is often the case if you register in a window open event and use the plugin in a control open event.
On the console on Mac OS X or Windows, you may see a message like this "MBS Plugins were used by the application before the RegisterMBSPlugin function was called. Please fix this in your code!".

2. you may have mixed different plugin versions which are not compatible.
In this case you can see a message "Internal plugin registration error." on the console on Mac OS X. Newer plugins may show a message dialog reporting this. Older version simply think they are not registered.

If the installer just merges old and new applications, users may have libraries of older and newer plugin versions in the libs folder. If your application loads the wrong version, the registration fails.

If you use remote debugging, make sure you clear the tempory files there, too. Otherwise you may have old DLLs on your hard disc which may disturb your application.

You can run into issues if you use your registration code on different places of your app. Please register only once in app.open (or app Constructor). If you have several codes, simply call them one after the other.

Also check that you only call RegisterMBSPlugin with valid serial number. If you later call RegisterMBSPlugin with Demo like in example code above, you remove the license.

Next check if you can clear the Xojo caches and that helps. This includes the Xojo Scratch folder and the Plugins & Project caches. Simply locate those folders and delete them. For Windows look in hidden AppData folder in your user folder. For Mac, please check
textasciitilde /Library/Caches and your temp folders.

Finally make sure you use the right serial number. Not an older one or a misspelled one.

20.0.215 I want to accept Drag & Drop from iTunes

Plugin Version: all, Platform: macOS.

Answer: You need to accept AcceptMacDataDrop "itun" and Handle the DropObject.
Example:

```
Sub Open()
window1.AcceptMacDataDrop "itun"
End Sub

Sub DropObject(obj As DragItem)
dim s as string
dim f as folderItem
dim d as CFDictionaryMBS
dim o as CFObjectMBS
dim key as CFStringMBS
dim dl as CFDictionaryListMBS
dim i,c as Integer
dim u as CFURLMBS
dim file as FolderItem

if obj.MacDataAvailable("itun") then
s = obj.MacData("itun")
```

```
// Parse XML
o=NewCFObjectMBSFromXML(NewCFBinaryDataMBSStr(s))

// Make dictionary
if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)

// get Tracks Dictionary
key=NewCFStringMBS("Tracks")
o=d.Value(key)

if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)
dl=d.List

// Walk over all entries in the Tracks dictionary
c=dl.Count-1
for i=0 to c
o=dl.Value(i)

if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)

key=NewCFStringMBS("Location")
o=d.Value(key)
if o isa CFStringMBS then
u=NewCFURLMBSCFStringMBS(CFStringMBS(o),nil)

file=u.file
if file<>nil then
MsgBox file.NativePath
end if
end if
end if
next
end if
end if
end if
End Sub
```

Notes: The code above inside a window on Xojo 5.5 with MBS Plugin 5.3 will do it nice and show the paths.

20.0.216 I'm drawing into a listbox but don't see something.

Plugin Version: all.

Answer: If you draw this in a listbox cellbackground, you need to draw on the correct position
Example:

```
Function CellBackgroundPaint(g As Graphics, row as Integer, column as Integer) As Boolean
dim f as FolderItem
f=SpecialFolder.Desktop
f.DrawWideIconMBS(g,listbox1.left,listbox1.top+row*20,16)
Return true
End Function
```

Notes: Try this in a listbox. The Graphics object there has a cliping and an offset which the plugin doesn't know about.

20.0.217 I'm searching for a method or so to move a window from position x.y to somewhere else on the screen.

Platform: macOS.

Answer:

The code I produced in RB isn't smooth enough. Is there a call in MBS, if not, can it be done? The speed of it has to be like the show of a DrawerWindow.

Try the declare below for Carbon. With WindowLib it will work on Mac OS 8.5 and newer.
Notes:

See Window.Transition functions.

20.0.218 If I use one of your plug-ins under windows, would this then impose the use of dll after compilation or my would my compiled soft still be a stand-alone single file software?

Platforms: macOS, Linux, Windows.

Answer: Stand alone.
Notes: Xojo compiles all used plugins into the application binary.
Some plugin parts need external dlls but you will find that in the documentation. (e.g. pdflib for some classes)

20.0.219 Is the fn key on a powerbook keyboard down?

Plugin Version: all, Platform: macOS.

Answer: I am unable to figure out how or if it is possible to detect if the fn key is down on a powerbook keyboard. Is it possible?
Example:

```
' Window.Open Event of a blank project:

dim i as Integer

for i=0 to 127
if keyboard.asynckeydown(i) then
title=str(i) // found
return
end if
next
title="" // not found
```

Notes: This test application shows the keycode (decimal) 63 for the fn key.

20.0.220 Is there a case sensitive Dictionary?

Plugin Version: all.

Answer: The MBS Plugin has several classes which can work as a replacement.
Notes: First you could use VariantToVariantHashMapMBS or VariantToVariantOrderedMapMBS.
If you know that all keys are Strings or Integers only, you can use the specialized classes which are a little bit faster due to avoiding variants:

IntegerToIntegerHashMapMBS class
IntegerToIntegerOrderedMapMBS class
IntegerToStringHashMapMBS class
IntegerToStringOrderedMapMBS class
IntegerToVariantHashMapMBS class
IntegerToVariantOrderedMapMBS class
StringToStringHashMapMBS class
StringToStringOrderedMapMBS class
StringToVariantHashMapMBS class
StringToVariantOrderedMapMBS class

20.0.221 Is there a way to use the MBS plugin to get only the visible item and folder count on a volume?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the DirectorySizeMBS class for this as in the example below:
Example:

```
dim d as DirectorySizeMBS

d=new DirectorySizeMBS

// volume(1) as my boot volume is very full
if d.update(volume(1),true,0) then
MsgBox str(d.VisibleItemCount)+" visible items, "+str(d.HiddenItemCount)+" invisible items."
end if
```

Notes: Complete Question: Is there a way to use the MBS plugin to get only the visible item and folder count on a volume? The FileCount and FolderCount properties of VolumeInformationMBS seem to provide the total # of items including invisible items such as .DS_Store and more importantly .Trashes which is causing me a great amount of difficulty during a recursive scan of a volume. I've got a progress bar which uses the total of the filecount and foldercount properties as the maximum value, but my routine needs to filter out all invisible items, as it is creating a catalog of a volume for archiving purposes. Any thoughts how I could get accurate number.

20.0.222 Is there an easy way I can launch the Displays preferences panel?

Plugin Version: all, Platform: macOS.

Answer: Use the code below:
Example:

```
dim error as Integer

error=OpenMacOSXPreferencesPaneMBS("Displays")
if error<>0 then
MsgBox "Failed to launch QuickTime System Preferences panel."
end if
```

20.0.223 List of Windows Error codes?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have a list of windows error codes on our website.
Notes: http://www.monkeybreadsoftware.de/xojo/winerror.shtml

20.0.224 Midi latency on Windows problem?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The issue is system related, not a problem with RB or the plugin.
Notes: Two things will adversely affect the timing:

(1) latency of the software synthesizer output driver. The default Windows wavetable synthesizer has considerable latency. I don't know how many milliseconds, but it is noticeable.

(2) latency of the digital audio output driver. Different systems have different drivers for different audio hardware. My Dell laptop has a minimum 15ms latency in the audio driver.

These two things put together were causing a very sluggish MIDI response. I was able to verify these as the culprits by routing MIDI directly out of RB into a sample player, which only introduces the latency of (2) and does not include latency of (1).

I don't know how widely known are these facts, if not then you may want to add this information to the documentation, since Windows programmers using the MIDI plugin may not know those problems, and might mistakenly blame your plugin, as I did :) Sorry about that!

(From Aaron Andrew Hunt)

20.0.225 My Xojo Web App does not launch. Why?

Plugin Version: all, Platform: macOS.

Answer: Here is a list of checks to do for linux apache installations with Xojo or Xojo Web applications:
Notes: Just a list of checks to do for linux apache installations:

- You have 64bit linux? Than you need 32 bit compatibility libraries.
- The folder of your app is writable? Set permissions to 777.
- The cgi script is executable? Set permissions to 755.

- The app file itself is executable? Set permissions to 755.
- You uploaded cgi file as text, so it has unix line endings? (this often gives error ”Premature end of script headers” in apache log)
- You uploaded config.cfg file and made it writable? Set permissions to 666.
- Your apache allows execution of cgi scripts? You enabled cgi for apache and uncommented addhandler command for CGI on a new apache installation?
- You uploaded the app file and libraries as binary files? Upload as text breaks them.
- You did upload the libs folder?
- You don’t have code in app.open, session.open and other events which crashes app right at launch?
- You don”t have a print command in your app.open event? (see feedback case 23817)
- You allowed htaccess file to overwrite permissions?

20.0.226 SQLDatabase not initialized error?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Before you can use SQLDatabaseMBS, it must be initialized.
Example:

```
dim d as new SQLDatabaseMBS
```

Notes: This happens normally when you use ”new SQLDatabaseMBS”.
But if you just have a SQLConnectionMBS and get a recordset there, the initialization may not have happend, yet.
So please simply add a line ”dim d as new SQLDatabaseMBS” to your app.open code after registration, so the plugin part can initialize and late provide recordsets.

20.0.227 Textconverter returns only the first x characters. Why?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Some older Xojo versions limit the Textconverter to around 1024 characters in input and output.
This should be fixed with RB5.
Notes:

Xojo seems not to support Textconverters at all on Windows.

20.0.228 The type translation between CoreFoundation/Foundation and Xojo data types.

Plugin Version: all, Platform: macOS.

Answer: The plugin does conversion between Cocoa/Carbon data types and native Xojo data types. The following list help you knowing what the current plugins support:
Notes: Cocoa NSObject to Variant:

nil ->nil
NSDictionary ->Dictionary
NSData ->MemoryBlock
NSString ->String
NSAttributedString ->NSAttributedStringMBS
NSDate ->Date
NSNumber ->double/integer/Int64/UInt64/UInt32/Boolean
NSURL ->String
NSValue with NSRect ->NSRectMBS
NSValue with NSPoint ->NSPointMBS
NSValue with NSSize ->NSSizeMBS
NSValue with NSRange ->NSRangeMBS
NSValue with QTTime ->QTTimeMBS
NSValue with QTTimeRange ->QTTimeRangeMBS
NSArray ->Array of Variant
QuartzFilter ->QuartzFilterMBS

- ->*MBS

Variant to Cocoa NSObject:

nil ->nil
Dictionary ->NSDictionary
Boolean ->NSNumber
Integer ->NSNumber
Color ->NSColor
Int64 ->NSNumber
Single ->NSNumber
Double ->NSNumber
Date ->NSDate
MemoryBlock ->NSData
String ->NSString
NSImageMBS ->NSImage
NSAttributedStringMBS ->NSAttributedString
NSColorMBS ->NSColor
NSRectMBS ->NSValue with NSRect
NSSizeMBS ->NSValue with NSSize

NSPointMBS ->NSValue with NSPoint
NSRangeMBS ->NSValue with NSRange
NSBurnMBS ->NSBurn
NSViewMBS ->NSView
NSFontMBS ->NSFont
NSParagraphStyleMBS ->NSParagraphStyle
NSAttributedStringMBS ->NSAttributedString
WebPolicyDelegateMBS ->WebPolicyDelegate
WebUIDelegateMBS ->WebUIDelegate
WebFrameLoadDelegateMBS ->WebFrameLoadDelegate
WebResourceLoadDelegateMBS ->WebResourceLoadDelegate
NSIndexSetMBS ->NSIndexSet
QTTimeMBS ->QTTime
QTTimeRangeMBS ->QTTimeRange
Array of Variant ->NSArray
Array of String ->NSArray
CFStringMBS ->NSString
CFNumberMBS ->NSNumber
CFDataMBS ->NSData
CFURLMBS ->NSURL
CFArrayMBS ->NSArray
CFDictionaryMBS ->NSDictionary
CFBinaryDataMBS ->NSDate

Carbon CFTypeRef to Variant:

CFDictionaryRef ->Dictionary
CFStringRef ->String
CFDataRef ->String
CFURL ->String
CFNumber ->Integer/Double/Int64
CFArray ->Array
CFDate ->date
nil ->nil
CGColorSpace ->CGColorSpaceMBS
CGColor ->CGColorMBS
CGImage ->CGImageMBS
CF* ->CF*MBS

Variant to Carbon CFTypeRef:

Dictionary ->CFDictionaryRef
Boolean ->CFBooleanRef
Color ->CFNumberRef
Integer ->CFNumberRef

Int64 ->CFNumberRef
Single ->CFNumberRef
Double ->CFNumberRef
String ->CFStringRef
Color ->CGColorRef
Date ->CFDateRef
nil ->nil
Memoryblock ->CFDataRef
FolderItem ->CFURLRef
Dictionary ->CFDictionaryRef
Array of Variant/String/Date/Double/Single/Int64/Integer ->CFArray
CGRectMBS ->CGRect as CFDataRef
CGSizeMBS ->CGSize as CFDataRef
CGPointMBS ->CGPoint as CFDataRef
CGColorMBS ->CGColor
CGColorSpaceMBS ->CGColorSpace
CGImageMBS ->CGImage
CGDataConsumerMBS ->CGDataConsumer
CGDataProviderMBS ->CGDataProvider
CF*MBS ->CF*

Strings without encodings should be put into dictionaries as memoryblocks.

20.0.229 Uploaded my web app with FTP, but it does not run on the server!

Plugin Version: all, Platform: Windows.

Answer: If you see errors like a simple "Segmentation Fault" on Linux or some other wired errors, you may want to check your FTP upload mode. It must be binary for web apps. ASCII mode corrupts the application.

20.0.230 What classes to use for hotkeys?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use CarbonHotKeyMBS class on Mac and WindowsKeyFilterMBS on Windows.
Notes: CarbonHotKeyMBS will also work fine in Cocoa apps.

20.0.231 What do I need for Linux to get picture functions working?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In order to get our plugins working on Linux systems without GUI, the plugin loads graphics

libraries dynamically.
Notes: To get it working, the plugin tries to load gtk with this paths:

- libgtk-x11-2.0.so"
- libgtk-x11-2.0.so.0"
- /usr/lib/libgtk-x11-2.0.so"
- /usr/lib32/libgtk-x11-2.0.so"
- /usr/lib/libgtk-x11-2.0.so.0"
- /usr/lib32/libgtk-x11-2.0.so.0"

gdk is loaded with this paths:

- libgdk-x11-2.0.so"
- libgdk-x11-2.0.so.0"
- /usr/lib/libgdk-x11-2.0.so"
- /usr/lib32/libgdk-x11-2.0.so"
- /usr/lib/libgdk-x11-2.0.so.0"
- /usr/lib32/libgdk-x11-2.0.so.0"

For the paths without explicit path, the system will search in /lib, /usr/lib and all directories in the LD_LIBRARY_PATH environment variable.

20.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

20.0.233 What font is used as a 'small font' in typical Mac OS X apps?

Plugin Version: all, Platform: macOS.

Answer:

Xojo 4.5 has a constant "SmallSystem" to use for a font name.
For older versions try this code:
Example:

```
Sub GetThemeFont(fontType as Integer, ByRef fontName as String, ByRef fontSize as Integer, ByRef
fontStyle as Integer)
dim err as Integer
dim theFont, theFontSize, theFontStyle as MemoryBlock

const smSystemScript = -1

Declare Function GetThemeFont Lib "Carbon" (inFontID as Integer, inScript as Integer, outFontName
as Ptr, outFontSize as Ptr, outStyle as Ptr) as Integer

theFont = NewMemoryBlock(256) //Str255
theFontSize = NewMemoryBlock(2) //SInt16
theFontStyle = NewMemoryBlock(1) //Style

err = GetThemeFont(fontType, smSystemScript, theFont, theFontSize, theFontStyle)

if err = 0 then
fontName = theFont.PString(0)
fontSize = theFontSize.UShort(0)
fontStyle = theFontStyle.Byte(0)
else
fontName = ""
fontSize = 0
fontStyle = 0
end if
End Sub
```

20.0.234 What is last plugin version to run on Mac OS X 10.4?

Plugin Version: all, Platform: Windows.

Answer: Last Version with 10.4 support is version 15.4.
Notes: With version 15.4 you can build applications for OS X 10.4 and newer.
For Version 16.0 we disabled 10.4 and moved minimum to 10.5. We may be able to enable it again to build a version of 16.x, but may need to charge for this by hour.

20.0.235 What is last plugin version to run on PPC?

Plugin Version: all, Platform: Windows.

Answer: Last Version with PPC is 15.4.
Notes: With version 15.4 you can build PPC applications for OS X 10.4 and newer.
For Version 16.0 we disabled PPC. We may be able to enable it again to build a PPC version of 16.x, but may need to charge for this by hour.

20.0.236 What is last version of the plugins for macOS 32-bit?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use version 23.0 or older.
Notes: We stopped including 32-bit code for macOS in version 23.1.
Please us older versions if you use an old Xojo.

Xojo 2017r3 and newer load our 64-bit plugins.

20.0.237 What is the difference between Timer and WebTimer?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Time is server side and WebTimer client side.
Notes: Timer is the normal timer class in Xojo. It runs on the server. On the side the WebTimer runs on the client. It triggers a request to the server to perform the action. So a WebTimer is good to keep the connection running and the website updated regularly. A timer on the server is good to make regular jobs like starting a database backup every 24 hours.

20.0.238 What is the list of Excel functions?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below a list of function names known by LibXL.
Notes: LibXL parses the functions and writes tokens to the excel file. So even if Excel can do more functions, we can only accept the ones known by LibXL.

ABS, ABSREF, ACOS, ACOSH, ACTIVE.CELL, ADD.BAR, ADD.COMMAND, ADD.MENU, ADD.TOOLBAR, ADDRESS, AND, APP.TITLE, AREAS, ARGUMENT, ASC, ASIN, ASINH, ATAN, ATAN2, ATANH, AVEDEV, AVERAGE, AVERAGEA, BAHTTEXT, BETADIST, BETAINV, BINOMDIST, BREAK, CALL, CALLER, CANCEL.KEY, CEILING, CELL, CHAR, CHECK.COMMAND, CHIDIST, CHIINV, CHITEST, CHOOSE, CLEAN, CODE, COLUMN, COLUMNS, COMBIN, CONCATENATE, CONFIDENCE, CORREL, COS, COSH, COUNT, COUNTA, COUNTBLANK, COUNTIF, COVAR, CREATE.OBJECT, CRITBINOM, CUSTOM.REPEAT, CUSTOM.UNDO, DATE, DATEDIF, DATESTRING, DATEVALUE, DAVERAGE, DAY, DAYS360, DB, DBCS, DCOUNT, DCOUNTA, DDB, DEGREES, DELETE.BAR, DELETE.COMMAND, DELETE.MENU, DELETE.TOOLBAR, DEREF, DEVSQ, DGET, DIALOG.BOX, DIRECTORY, DMAX, DMIN, DOCUMENTS, DOLLAR, DPRODUCT, DSTDEV, DSTDEVP, DSUM, DVAR, DVARP, ECHO, ELSE, ELSE.IF, ENABLE.COMMAND, ENABLE.TOOL, END.IF, ERROR, ERROR.TYPE, EVALUATE, EVEN, EXACT, EXEC, EXECUTE, EXP, EXPONDIST, FACT, FALSE, FCLOSE, FDIST, FILES, FIND, FINDB, FINV, FISHER, FISHERINV, FIXED, FLOOR, FOPEN, FOR, FOR.CELL, FORECAST,

FORMULA.CONVERT, FPOS, FREAD, FREADLN, FREQUENCY, FSIZE, FTEST, FV, FWRITE, FWRITELN, GAMMADIST, GAMMAINV, GAMMALN, GEOMEAN, GET.BAR, GET.CELL, GET.CHART.ITEM, GET.DEF, GET.DOCUMENT, GET.FORMULA, GET.LINK.INFO, GET.MOVIE, GET.NAME, GET.NOTE, GET.OBJECT, GET.PIVOT.FIELD, GET.PIVOT.ITEM, GET.PIVOT.TABLE, GET.TOOL, GET.TOOLBAR, GET.WINDOW, GET.WORKBOOK, GET.WORKSPACE, GETPIVOTDATA, GOTO, GROUP, GROWTH, HALT, HARMEAN, HELP, HLOOKUP, HOUR, HYPERLINK, HYPGEOMDIST, IF, INDEX, INDIRECT, INFO, INITIATE, INPUT, INT, INTERCEPT, IPMT, IRR, ISBLANK, ISERR, ISERROR, ISLOGICAL, ISNA, ISNONTEXT, ISNUMBER, ISPMT, ISREF, ISTEXT, ISTHAIDIGIT, KURT, LARGE, LAST.ERROR, LEFT, LEFTB, LEN, LENB, LINEST, LINKS, LN, LOG, LOG10, LOGEST, LOGINV, LOGNORMDIST, LOOKUP, LOWER, MATCH, MAX, MAXA, MDETERM, MEDIAN, MID, MIDB, MIN, MINA, MINUTE, MINVERSE, MIRR, MMULT, MOD, MODE, MONTH, MOVIE.COMMAND, N, NA, NAMES, NEGBINOMDIST, NEXT, NORMDIST, NORMINV, NORMSDIST, NORMSINV, NOT, NOTE, NOW, NPER, NPV, NUMBERSTRING, ODD, OFFSET, OPEN.DIALOG, OPTIONS.LISTS.GET, OR, PAUSE, PEARSON, PERCENTILE, PERCENTRANK, PERMUT, PHONETIC, PI, PIVOT.ADD.DATA, PMT, POISSON, POKE, POWER, PPMT, PRESS.TOOL, PROB, PRODUCT, PROPER, PV, QUARTILE, RADIANS, RAND, RANK, RATE, REFTEXT, REGISTER, REGISTER.ID, RELREF, RENAME.COMMAND, REPLACE, REPLACEB, REPT, REQUEST, RESET.TOOLBAR, RESTART, RESULT, RESUME, RETURN, RIGHT, RIGHTB, ROMAN, ROUND, ROUNDBAHTDOWN, ROUNDBAHTUP, ROUNDDOWN, ROUNDUP, ROW, ROWS, RSQ, RTD, SAVE.DIALOG, SAVE.TOOLBAR, SCENARIO.GET, SEARCH, SEARCHB, SECOND, SELECTION, SERIES, SET.NAME, SET.VALUE, SHOW.BAR, SIGN, SIN, SINH, SKEW, SLN, SLOPE, SMALL, SPELLING.CHECK, SQRT, STANDARDIZE, STDEV, STDEVA, STDEVP, STDEVPA, STEP, STEYX, SUBSTITUTE, SUBTOTAL, SUM, SUMIF, SUMPRODUCT, SUMSQ, SUMX2MY2, SUMX2PY2, SUMXMY2, SYD, T, TAN, TANH, TDIST, TERMINATE, TEXT, TEXT.BOX, TEXTREF, THAIDAYOFWEEK, THAIDIGIT, THAIMONTHOFYEAR, THAINUMSOUND, THAINUMSTRING, THAISTRINGLENGTH, THAIYEAR, TIME, TIMEVALUE, TINV, TODAY, TRANSPOSE, TREND, TRIM, TRIMMEAN, TRUE, TRUNC, TTEST, TYPE, UNREGISTER, UPPER, USDOLLAR, USERDEFINED, VALUE, VAR, VARA, VARP, VARPA, VDB, VIEW.GET, VLOOKUP, VOLATILE, WEEKDAY, WEIBULL, WHILE, WINDOW.TITLE, WINDOWS, YEAR and ZTEST.

20.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

20.0.240 What to do on Xojo reporting a conflict?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

I get an error like ”This item conflicts with another item of the same name” when using one of the plugin functions.

Xojo just wants to tell you that you dropped something in the plugins folder what is not a plugin.
Notes:

Some users dropped the examples, the documentation or other files into the plugins folder. Don't do it.

20.0.241 What to do with a NSImageCacheException?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You need to add exception handlers for NSExceptionMBS in order to catch this exception.
Notes: You may also add code to write the stack of the exception into a log file for later locating the error source.

A NSImage has several image representations in memory. So basicly you pass in the base image and for whatever size an image is needed, the NSImage class will create a cache image representation of the requested size so on the next query it can use that cache for the same requested size.

20.0.242 What to do with MySQL Error 2014?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can get this error on MySQL if you have a recordset open while you create another one.

20.0.243 What to do with SQL Plugin reporting Malformed string as error?

Plugin Version: all, Platform: macOS.

Answer: Please make sure the table and/or database fields have a text encoding set.
Notes: For Firebird our plugin tries to use UTF-8 encoding if possible and to correctly convert between various tables, the tables and their fields need to have a text encoding defined.
e.g. if the text field in the table is windows-1252 and the other ISO 8859-5, then the Firebird database can convert them to UTF-8 and deliver texts to the plugin.
If encoding is set to none, it may get confused for non-ascii text.

20.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

20.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

20.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

20.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

20.0.248 Where is GetObjectClassNameMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use this replacement method:
Example:

```
Function GetObjectClassNameMBS(o as Object) As string
dim t as Introspection.TypeInfo = Introspection.GetType(o)
Return t.FullName
End Function
```

Notes: GetObjectClassNameMBS was removed from the plugins.

20.0.249 Where is NetworkAvailableMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We removed NetworkAvailableMBS some versions ago. It was not working right and basicly it's not useful. If you want to check whether you have a network, than do a DNS resolve:
Example:

```
// two independend domain names
const domain1 = "www.google.com"
const domain2 = "www.macsw.de"

// resolve IPs
dim ip1 as string = DNSNameToAddressMBS(Domain1)
dim ip2 as string = DNSNameToAddressMBS(Domain2)

// if we got IPs and not the same IPs (error/login pages)
if len(ip1)=0 or len(ip2)=0 or ip1=ip2 then
MsgBox "no connection"
else
MsgBox "have connection"
end if
```

Notes: This way you can detect whether you got something from DNS. And you can make sure that a DNS redirection to a login page won't catch you.

20.0.250 Where is StringHeight function in DynaPDF?

Plugin Version: all, Platform: Windows.

Answer: Use the function GetFTextHeight or GetFTextHeightEx.
Notes: Be aware that GetFTextHeight works with format commands and you may want to escape your text if you don't use them.

20.0.251 Where is XLSDocumentMBS class?

Plugin Version: all, Platform: macOS.

Answer: This class has been removed in favor of XLBookMBS class.
Notes: This classes have been removed XLSCellMBS, XLSDocumentMBS, XLSFormatRecordMBS, XLSMergedCellsMBS, XLSRowMBS and XLSSheetMBS.

20.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:
http://www.wotsit.org

20.0.253 Where to register creator code for my application?

Plugin Version: all, Platform: macOS.

Answer:

Register at Apple:
http://developer.apple.com/dev/cftype/information.html

20.0.254 Which Mac OS X frameworks are 64bit only?

Plugin Version: all, Platform: macOS.

Answer: Some frameworks from Mac OS X do not support 32 bit applications, so we can't provide plugins for Xojo until 64bit target is available.
Notes: For Mac OS X 10.8:

- Accounts
- EventKit
- GLKit
- Social

and in 10.9:

- Accounts
- AVKit
- EventKit
- GameController
- GLKit
- MapKit
- MediaLibrary
- Social
- SpriteKit

In general Apple makes all new frameworks being 64 bit only.

20.0.255 Which plugins are 64bit only?

Plugin Version: all, Platform: macOS.

Answer: Some of our plugins work only in 64 bit modes as operation systems do not provide 32 bit code.
Notes: This effects currently: EventKit, Accounts, Social frameworks from Apple and our matching plugins.

20.0.256 Why application doesn't launch because of a missing ddraw.dll!?

Plugin Version: all, Platform: Windows.

Answer: Some RB versions require that you install DirectX from Microsoft on your Windows.

20.0.257 Why application doesn't launch because of a missing shlwapi.dll!?

Plugin Version: all, Platform: Windows.

Answer: Some RB versions require that you install the Internet Explorer from Microsoft on your Windows.
Notes: This bug is for several older Windows 95 editions.

20.0.258 Why do I hear a beep on keydown?

Plugin Version: all, Platform: Windows.

Answer: When the user presses a key, RB goes through all keydown event handlers till on returns true.
Notes: If no keydown event handler returns true for the key, a beep is performed.

20.0.259 Why does folderitem.item return nil?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Because Xojo fails to make a folderitem for you. Reason may be an alias file which can't be resolved or simply that you don't have enough access rights to read the folder content.
Notes: A more rarely reason is that the directory changed and the file with the given index or name does no longer exist.

20.0.260 Why doesn't showurl work?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

There are three main reasons:

1. showurl is not supported by Xojo in 68k applications.
2. there is now application defined for the protocol (e.g. http) in the Internet Control panel.
3. You don't have Internet Config installed.

You can use the InternetConfigMBS class to check for this stuff.

20.0.261 Why don't the picture functions not work on Linux?

Plugin Version: all, Platform: macOS.

Answer: Please make sure libcairo is installed.
Notes: For accessing pictures on Linux, the MBS Plugin relays on the cairo library.
Please install the package if you don't have it already.
Our plugin looks for library called libcairo.so or libcairo.so.2.

20.0.262 Why have I no values in my chart?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You have no data points visible, there may be several reasons:
Notes: For example one of the data values may be infinite or invalid.
Or the scaling may be out of range, so you simply see nothing.

20.0.263 Will application size increase with using plugins?

Plugin Version: all, Platform: Windows.

Answer: All plugins used by your application will be included in the application.
Notes: If you use no plugins, your application will not change size.
And if you use one class from the plugins, your application size will increase by a few kilobytes.
The documentation of the plugins include a list of all plugin parts and their sizes for the different platforms.

20.0.264 XLS: Custom format string guidelines

Plugin Version: all, Platform: macOS.

Answer: You have to download the source code and compile a static version of the library.
Notes: Up to four sections of format codes can be specified. The format codes, separated by semicolons, define the formats for positive numbers, negative numbers, zero values, and text, in that order. If only two sections are specified, the first is used for positive numbers and zeros, and the second is used for negative numbers. If only one section is specified, it is used for all numbers. Four sections example:

#,###.00_); [Red] (#,###.00);0.00;"sales "@

The following table describes the different symbols that are available for use in custom number formats.

Specify colors

To set the text color for a section of the format, type the name of one of the following eight colors in square brackets in the section. The color code must be the first item in the section.

Instead of using the name of the color, the color index can be used, like this [Color3] for Red. Valid numeric indexes for color range from 1 to 56, which reference by index to the legacy color palette.
Specify conditions

To set number formats that will be applied only if a number meets a specified condition, enclose the condition in square brackets. The condition consists of a comparison operator and a value. Comparison operators include: = Equal to; >Greater than; <Less than; >= Greater than or equal to, <= Less than or equal to, and <>Not equal to. For example, the following format displays numbers that are less than or equal to 100 in a red font and numbers that are greater than 100 in a blue font.
[Red] [<=100] ; [Blue] [>100]
If the cell value does not meet any of the criteria, then pound signs ("#") are displayed across the width of the cell.
Dates and times

Examples

20.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

20.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic. Why?

Plugin Version: all.

Answer:

You may check if the application has enough memory to be loaded.
RB should have on Mac OS Classic more than 20 MB of RAM.
I prefered to use 50 MB and for an application a 10 MB partition is a good way to start.

Parameter	Description
x	The x value of the data point. For an enumerated x-axis (see Axis.setLabels on what is an enumerated axis), the first data point is 0, and the nth data point is (n-1).
xLabel	The bottom x-axis label of the data point.
x2Label	The top x-axis label of the data point.
value	The value of the data point.
accValue	The sum of values of all data points that are in the same x position and same data group as the current data point, and with data set number less than or equal to the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
totalValue	The sum of values of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
percent	The percentage of the data point based on the total value of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
accPercent	The accumulated percentage of the data point based on the total value of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
gpercent	The percentage of the data point based on the total value of all data points in a layer.
dataSet	The data set number to which the data point belongs. The first data set is 0. The nth data set is (n-1).
dataSetName	The name of the data set to which the data point belongs.
dataItem	The data point number within the data set. The first data point is 0. The nth data point is (n-1).
dataGroup	The data group number to which the data point belongs. The first data group is 0. The nth data group is (n-1).
dataGroupName	The name of the data group to which the data point belongs.
layerId	The layer number to which the data point belongs. The first layer is 0. The nth layer is (n-1).
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using Layer.addExtraField, Layer.addExtraField2, BaseChart.addExtraField or BaseChart.addExtraField2.

diFieldN	Same as fieldN. See above.
dsFieldN	Similar to fieldN, except that dsFieldN means the extra field is indexed by data set number. The Pth data set corresponds to the Pth element of the extra field.
dsdiFieldN	Similar to fieldN, except that dsdiFieldN means the extra fields are indexed by both the data set number and data point number. The Pth data item of the Qth data set corresponds to the Pth element of the (N + Q)th extra field.

Parameter	Description
zx	The symbol scale in the x dimension. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.
zy	The symbol scale in the y dimension. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.
z	The symbol scale without distinguishing the dimension to use. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.

Parameter	Description
slope	The slope of the trend line.
intercept	The y-intercept of the trend line.
corr	The correlation coefficient in linear regression analysis.
stderr	The standard error in linear regression analysis.

Parameter	Description
top	The value of the top edge of the box-whisker symbol.
bottom	The value of the bottom edge of the box-whisker symbol.
max	The value of the maximum mark of the box-whisker symbol.
min	The value of the minimum mark of the box-whisker symbol.
med	The value of the median mark of the box-whisker symbol.

Parameter	Description
high	The high value.
low	The low value.
open	The open value.
close	The close value.

Parameter	Description
dir	The direction of the vector.
len	The length of the vector.

Parameter	Description
radius	The radial value of the data point.
value	Same as { radius } . See above.
angle	The angular value of the data point.
x	Same as { angle } . See above.
label	The angular label of the data point.
xLabel	Same as { label } . See above.
name	The name of the layer to which the data point belongs.
dataSetName	Same as { name } . See above.
i	The data point number. The first data point is 0. The nth data point is (n-1).
dataItem	Same as { i } . See above.
z	The symbol scale. Applicable for layers with symbol scales set by PolarLayer.setSymbolScale.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using Layer.addExtraField, Layer.addExtraField2, BaseChart.addExtraField or BaseChart.addExtraField2.

diFieldN	Same as fieldN. See above.
dsFieldN	Similar to fieldN, except that dsFieldN means the extra field is indexed by layer index. The Pth layer corresponds to the Pth element of the extra field.
dsdiFieldN	Similar to fieldN, except that dsdiFieldN means the extra fields are indexed by both the data set number and data point number. The Pth data item of the Qth layer corresponds to the Pth element of the (N + Q)th extra field.

Parameter	Description
dir	The direction of the vector.
len	The length of the vector.

Parameter	Description
value	The axis value at the tick position.
label	The axis label at the tick position.

Parameter	Description
[param]	The name of the parameter
[a]	If this field a number, it specifies the number of decimal places (digits to the right of the decimal point).

[b] The thousand separator. Should be a non-alphanumeric character (not 0-9, A-Z, a-z). Use ' textasciitilde ' for no thousand separator. The default is ' textasciitilde ', which can be modified using BaseChart.setNumberFormat.

[c] The decimal point character. The default is '.', which can be modified using BaseChart.setNumberFormat.

[d] The negative sign character. Use ' textasciitilde ' for no negative sign character. The default is '-', which can be modified using BaseChart.setNumberFormat.

Parameter	Description
yyyy	The year in 4 digits (e.g. 2002)
yyy	The year showing only the least significant 3 digits (e.g. 002 for the year 2002)
yy	The year showing only the least significant 2 digits (e.g. 02 for the year 2002)
y	The year showing only the least significant 1 digits (e.g. 2 for the year 2002)
mmm	The month formatted as its name. The default is to use the first 3 characters of the english month name (Jan, Feb, Mar ...). The names can be configured using BaseChart.setMonthNames.
mm	The month formatted as 2 digits from 01 - 12, adding leading zero if necessary.
m	The month formatted using the minimum number of digits from 1 - 12.
MMM	The first 3 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
MM	The first 2 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
M	The first character of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
dd	The day of month formatted as 2 digits from 01 - 31, adding leading zero if necessary.
d	The day of month formatted using the minimum number of digits from 1 - 31.
w	The name of the day of week. The default is to use the first 3 characters of the english day of week name (Sun, Mon, Tue ...). The names can be configured using BaseChart.setWeekDayNames.
hh	The hour of day formatted as 2 digits, adding leading zero if necessary. The 2 digits will be 00 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
h	The hour of day formatted using the minimum number of digits. The digits will be 0 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
nn	The minute formatted as 2 digits from 00 - 59, adding leading zero if necessary.
n	The minute formatted using the minimum number of digits from 00 - 59.
ss	The second formatted as 2 digits from 00 - 59, adding leading zero if necessary.
s	The second formatted using the minimum number of digits from 00 - 59.
a	Display either 'am' or 'pm', depending on whether the time is in the morning or afternoon. The text 'am' and 'pm' can be modified using BaseChart.setAMPM.

Shape Id	Value	Description
SquareShape	1	Square shape. See (1, 1) above.
DiamondShape	2	Diamond shape. See (2, 1) above.
TriangleShape	3	Triangle shape pointing upwards. See (3, 1) above.
RightTriangleShape	4	Triangle shape pointing rightwards. See (4, 1) above.
LeftTriangleShape	5	Triangle shape pointing leftwards. See (5, 1) above.
InvertedTriangleShape	6	Triangle shape pointing downwards. See (1, 2) above.
CircleShape	7	Circle shape. See (2, 2) above.
StarShape	[Method]	Star shapes of various points. See (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5) above for stars with 3 to 10 points.
PolygonShape	[Method]	Polygon shapes symmetrical about a vertical axis with a vertex at the top center position. See (4, 1), (4, 3), (4, 5), (5, 1) for polygons of 5 to 8 sides.
Polygon2Shape	[Method]	Polygon shapes symmetrical about a vertical axis but without any vertex at the top center position. See (4, 2), (4, 4) for polygons of 5 and 6 sides.
CrossShape	[Method]	'+' shapes. See (5, 2), (5, 3), (5, 4), (5, 5), (6, 1), (6, 2), (6. 3) for '+' shape with arm width of 0.1 - 0.7.
Cross2Shape	[Method]	'X' shapes. See (6, 4), (6, 5), (7, 1), (7, 2), (7, 3), (7, 4), (7, 5) for 'X' shapes with arm width of 0.1 - 0.7.

langEnglish	0	Roman script
langFrench	1	Roman script
langGerman	2	Roman script
langItalian	3	Roman script
langDutch	4	Roman script
langSwedish	5	Roman script
langSpanish	6	Roman script
langDanish	7	Roman script
langPortuguese	8	Roman script
langNorwegian	9	Roman script
langHebrew	10	Hebrew script
langJapanese	11	Japanese script
langArabic	12	Arabic script
langFinnish	13	Roman script
langGreek	14	Greek script using smRoman script code
langIcelandic	15	modified smRoman/Icelandic script
langMaltese	16	Roman script
langTurkish	17	modified smRoman/Turkish script
langCroatian	18	modified smRoman/Croatian script
langTradChinese	19	Chinese (Mandarin) in traditional characters
langUrdu	20	Arabic script
langHindi	21	Devanagari script
langThai	22	Thai script
langKorean	23	Korean script

Nan	Meaning
1	Invalid square root (negative number, usually)
2	Invalid addition (indeterminate such as infinity + (-infinity))
4	Invalid division (indeterminate such as 0/0)
8	Invalid multiplication (indeterminate such as 0*infinity)
9	Invalid modulo such as (a mod 0)
17	Try to convert invalid string to a number like val(”x7”)
33	Invalid argument in a trig function
34	Invalid argument in an inverse trig function
36	Invalid argument in a log function
37	Invalid argument in Pow function
38	Invalid argument in toolbox financial function
40	Invalid argument in hyperbolic function
42	Invalid argument in a gamma function

Symbol	Description and result
0	Digit placeholder. For example, if the value 8.9 is to be displayed as 8.90, use the format #.00
#	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall not display extra zeros when the number typed has fewer digits on either side of the decimal than there are # symbols in the format. For example, if the custom format is #.##, and 8.9 is in the cell, the number 8.9 is displayed.
?	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall put a space for insignificant zeros on either side of the decimal point so that decimal points are aligned in the column. For example, the custom format 0.0? aligns the decimal points for the numbers 8.9 and 88.99 in a column.
. (period)	Decimal point.
%	Percentage. If the cell contains a number between 0 and 1, and the custom format 0% is used, the application shall multiply the number by 100 and adds the percentage symbol in the cell.
, (comma)	Thousands separator. The application shall separate thousands by commas if the format contains a comma that is enclosed by number signs (#) or by zeros. A comma that follows a placeholder scales the number by one thousand. For example, if the format is #.0,, and the cell value is 12,200,000 then the number 12.2 is displayed.
E- E+ e- e+	Scientific format. The application shall display a number to the right of the "E" symbol that corresponds to the number of places that the decimal point was moved. For example, if the format is 0.00E+00, and the value 12,200,000 is in the cell, the number 1.22E+07 is displayed. If the number format is #0.0E+0, then the number 12.2E+6 is displayed.
$ -+/():space	Displays the symbol. If it is desired to display a character that differs from one of these symbols, precede the character with a backslash (\). Alternatively, enclose the character in quotation marks. For example, if the number format is (000), and the value 12 is in the cell, the number (012) is displayed.
\	Display the next character in the format. The application shall not display the backslash. For example, if the number format is 0\!, and the value 3 is in the cell, the value 3! is displayed.
*	Repeat the next character in the format enough times to fill the column to its current width. There shall not be more than one asterisk in one section of the format. If more than one asterisk appears in one section of the format, all but the last asterisk shall be ignored. For example, if the number format is 0*x, and the value 3 is in the cell, the value 3xxxxxx is displayed. The number of x characters that are displayed in the cell varies based on the width of the column.
_ (underline)	Skip the width of the next character. This is useful for lining up negative and positive values in different cells of the same column. For example, the number format _(0.0_);(0.0) aligns the numbers 2.3 and -4.5 in the column even though the negative number is enclosed by parentheses.
"text"	Display whatever text is inside the quotation marks. For example, the format 0.00 "dollars" displays 1.23 dollars when the value 1.23 is in the cell.
@	Text placeholder. If text is typed in the cell, the text from the cell is placed in the format where the at symbol (@) appears. For example, if the number format is "Bob "@" Smith" (including quotation marks), and the value "John" is in the cell, the value Bob John Smith is displayed.

[Black] [Green] [White] [Blue] [Magenta] [Yellow] [Cyan] [Red]

To display	As	Use this code
Months	1-12	m
Months	01-12	mm
Months	Jan-Dec	mmm
Months	January-December	mmmm
Months	J-D	mmmmm
Days	1-31	d
Days	01-31	dd
Days	Sun-Sat	ddd
Days	Sunday-Saturday	dddd
Years	00-99	yy
Years	1900-9999	yyyy
Hours	0-23	h
Hours	00-23	hh
Minutes	0-59	m
Minutes	00-59	mm
Seconds	0-59	s
Seconds	00-59	ss
Time	4 AM	h AM/PM
Time	4:36 PM	h:mm AM/PM
Time	4:36:03 P	h:mm:ss A/P
Time	4:36:03.75	h:mm:ss.00
Elapsed time	1:02	[h] :mm
Elapsed time	62:16	[mm] :ss
Elapsed time	3735.80	[ss] .00

To display	As	Use this code
1234.59	1234.6	####.#
8.9	8.900	#.000
.631	0.6	0.#
12	12.0	#.0#
1234.568	1234.57	#.0#
44.398	44.398	???.???
102.65	102.65	???.???
2.8	2.8	???.???
5.25	5 1/4	# ??/??
5.3	5 3/10	# ??/??
12000	12,000	#,###
12000	12	#,
12400000	12.4	0.0„