MBS AVFoundation 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 AVFoundation Plugin

0.2 Content

Chapter 1

List of Topics

Chapter 2

List of all classes

Chapter 3

List of all controls

• AVCaptureViewControlMBS	357
• AVPlayerViewControlMBS	614
• AVRoutePickerViewControlMBS	630
• DesktopAVCaptureViewControlMBS	721
• DesktopAVPlayerViewControlMBS	727
• DesktopAVRoutePickerViewControlMBS	736
• DesktopIKCameraDeviceViewControlMBS	921
• DesktopIKDeviceBrowserViewControlMBS	925
• DesktopIKImageBrowserViewControlMBS	1043
• DesktopIKImageViewControlMBS	1054
• DesktopIKScannerDeviceViewControlMBS	929
• IKCameraDeviceViewControlMBS	998
• IKDeviceBrowserViewControlMBS	1010
• IKImageBrowserViewControlMBS	1066
• IKImageViewControlMBS	1105
• IKScannerDeviceViewControlMBS	1019

Chapter 4

AVFoundation

4.1 class AVAssetExportSessionMBS

4.1.1 class AVAssetExportSessionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The AVFoundation class for media export.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim p as string = AVAssetExportSessionMBS.AVAssetExportPresetAppleM4VCellular
dim e as new AVAssetExportSessionMBS(a, p)

dim FileTypes() as string = e.supportedFileTypes
e.outputFileType = FileTypes(0)
e.OutputFile = SpecialFolder.Desktop.Child("small.m4v")
e.exportAsynchronously

// instead of waiting for event, we simply wait for it to finish
do
app.YieldToNextThread
loop until e.status <>e.AVAssetExportSessionStatusExporting
```

Notes: An AVAssetExportSession object transcodes the contents of an AVAsset source object to create an output of the form described by a specified export preset.

Prior to initializing an instance of AVAssetExportSession, you can use allExportPresets to get the complete list of presets available. Use exportPresetsCompatibleWithAsset to get a list of presets that are compatible with a specific asset.

After you have initialized an export session with the asset that contains the source media, the export preset name (presetName), and the output file type (outputFileType), you can start the export running by invoking exportAsynchronously. Because the export is performed asynchronously, this method returns immediately—you can use progress to check on the progress. Depending on the capabilities of the device, some exports may be queued when multiple exports are attempted. When this happens, the status of a queued export will indicate that it's waiting (AVAssetExportSessionStatusWaiting).

The exportAsynchronouslyCompleted event on AVFoundationMBS is called whether the export fails, completes, or is cancelled. Upon completion, the status property indicates whether the export has completed successfully. If it has failed, the value of the error property supplies additional information about the reason for the failure.
Available in OS X v10.7 and later.
Blog Entries

- Multithreaded plugin functions can increase speed of Xojo application
- MBS Xojo Plugins, version 18.4pr1
- MBS Xojo Plugins, version 18.2pr5
- Automated video editing with Xojo
- MBS Xojo / Real Studio Plugins, version 15.1pr1

4.1.2 Methods

4.1.3 allExportPresets as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns all available export preset names.
Notes: Returns an array containing a string constant for each of the available preset names.

4.1.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.1.5 AVAssetExportPreset1280x720 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: Specifies output at 1280x720 pixels.

4.1.6 AVAssetExportPreset1920x1080 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: Specifies output at 1920x1080 pixels.

4.1.7 AVAssetExportPreset3840x2160 as string

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: Specifies output at 3840x2160 pixels (4K video).

4.1.8 AVAssetExportPreset640x480 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: Specifies output at 640x480 pixels.

4.1.9 AVAssetExportPreset960x540 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: Specifies output at 960x540 pixels.

4.1.10 AVAssetExportPresetAppleM4A as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use this export option to produce an audio-only .m4a file with appropriate iTunes gapless playback data.
Notes: Specifies an audio-only .m4a file with appropriate iTunes gapless playback data.

4.1.11 AVAssetExportPresetAppleM4V1080pHD as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a 1080p High Definition format suitable for playing on Apple devices.

4.1.12 AVAssetExportPresetAppleM4V480pSD as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a 480p Standard Definition format suitable for playing on Apple devices.

4.1.13 AVAssetExportPresetAppleM4V720pHD as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a 720p High Definition format suitable for playing on Apple devices.

4.1.14 AVAssetExportPresetAppleM4VAppleTV as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a format suitable for playing on AppleTV.

4.1.15 AVAssetExportPresetAppleM4VCellular as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a format suitable for playing on Apple devices when streamed over a cellular network.

4.1.16 AVAssetExportPresetAppleM4ViPod as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a format suitable for playing on an iPod.

4.1.17 AVAssetExportPresetAppleM4VWiFi as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a format suitable for playing on Apple devices when streamed over a WiFi network.

4.1.18 AVAssetExportPresetAppleProRes422LPCM as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the export option presets to produce files that can be played on the specific Apple devices.
Notes: Specifies a QuickTime movie with Apple ProRes 422 video and LPCM audio.

4.1.19 AVAssetExportPresetHEVC1920x1080 as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with the specified video size.
The export will not scale the video up from a smaller size. The video will be compressed using HEVC and the audio will be compressed using AAC. Some devices cannot support some sizes.

4.1.20 AVAssetExportPresetHEVC3840x2160 as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with the specified video size.
The export will not scale the video up from a smaller size. The video will be compressed using HEVC and the audio will be compressed using AAC. Some devices cannot support some sizes.

4.1.21 AVAssetExportPresetHEVCHighestQuality as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with video size appropriate to the device. The export will not scale the video up from a smaller size. The video will be compressed using HEVC and the audio will be compressed using AAC.

4.1.22 AVAssetExportPresetHighestQuality as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with video size appropriate to the device. The export will not scale the video up from a smaller size. The video will be compressed using H.264 and the audio will be compressed using AAC.

4.1.23 AVAssetExportPresetLowQuality as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with video size appropriate to the device. The export will not scale the video up from a smaller size. The video will be compressed using H.264 and the audio will be compressed using AAC.

4.1.24 AVAssetExportPresetMediumQuality as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: You use these export options to produce QuickTime .mov files with a specified video size.
Notes: These export options can be used to produce movie files with video size appropriate to the device. The export will not scale the video up from a smaller size. The video will be compressed using H.264 and the audio will be compressed using AAC.

4.1.25 AVAssetExportPresetPassthrough as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use this export option to let all tracks pass through.
Notes: Specifies that all tracks pass through, unless it is not possible.

4.1.26 cancelExport

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels the execution of an export session.

4.1.27 Constructor(asset as AVAssetMBS, presetName as string)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset export session with a specified asset and preset.
Notes: asset: The asset you want to export.
presetName: A string constant specifying the name of the preset template for the export.

4.1.28 determineCompatibilityOfExportPreset(presetName as string, asset as AVAssetMBS, outputFileType as string, tag as Variant = nil)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Reports the compatibility of an export present, asset, and output file type to the event.
Notes: Calls later AVAssetExportSessionMBS.determineCompatibilityOfExportPresetCompleted event.

presetName: The name of the preset template for the export operation. For possible values, see ”Export Preset Names for Device-Appropriate QuickTime Files,” ”Export Preset Names for QuickTime Files of a Given Size,” AVAssetExportSessionStatusCancelled, ”Export Preset Name for iTunes Audio,” and ”Export Preset Name for Pass-Through.”
asset: The asset object that you are planning to export.
outputFileType: The UTI string corresponding to the file type. For example, to specify a QuickTime movie file format, you could specify the constant AVFileTypeQuickTimeMovie. For a list of constants specifying UTIs for standard file types, see AV Foundation Constants Reference.

Because not all export presets are compatible with all assets and file types, you can use this method to query the compatibility of specific combos before using them. To ensure that the export operation succeeds, you should not make any significant changes to the asset between the time of calling this method and performing the export operation.

This method performs its checks asynchronously on a secondary thread and returns immediately. The results are similarly reported to the event later.

Available in OS X v10.9 and later.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.1.29 determineCompatibleFileTypes(tag as Variant = nil)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Reports the compatible file types for the current export session to the event.
Notes: calls later AVFoundationMBS.determineCompatibleFileTypesCompleted event.

This method operates on the asset object and preset information that was used to initialize the export session. It uses this information to determine the file types that the export session can write.

This method performs its checks asynchronously on a secondary thread and returns immediately. The results are similarly reported to the specified block on a secondary thread. Because this method actually inspects the asset's tracks, the tracks are loaded if they have not been already.
Available in OS X v10.9 and later.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.1.30 exportAsynchronously(tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Starts the asynchronous execution of an export session.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim p as string = AVAssetExportSessionMBS.AVAssetExportPresetAppleM4VCellular
dim e as new AVAssetExportSessionMBS(a, p)

dim FileTypes() as string = e.supportedFileTypes
e.outputFileType = FileTypes(0)
e.OutputFile = SpecialFolder.Desktop.Child("small.m4v")
e.exportAsynchronously

// instead of waiting for event, we simply wait for it to finish
```

```
do
app.YieldToNextThread
loop until e.status <>e.AVAssetExportSessionStatusExporting
```

Notes: This method starts an asynchronous export operation and returns immediately. status signals the terminal state of the export session, and if a failure occurs, error describes the problem.

This method calls AVFoundationMBS.exportAsynchronouslyCompleted event later. Tag is passed so you can pass information needed to finish work.

If internal preparation for export fails, event is invoked synchronously. The event may also be called asynchronously, after the method returns, in the following cases:

- If a failure occurs during the export, including failures of loading, re-encoding, or writing media data to the output.
- If cancelExport is invoked.
- After the export session succeeds, having completely written its output to the outputURL.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.1.31 exportMT

Plugin Version: 18.2, Platform: macOS, Targets: All.

Function: Starts the synchronous execution of an export session.
Notes: Same as exportAsynchronously, but run synchronous, so the plugin waits.

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.

4.1.32 exportPresetsCompatibleWithAsset(asset as AVAssetMBS) as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the identifiers compatible with a given asset.
Notes: asset: An asset that is ready to be exported.

Returns an array of strings representing the identifiers compatible with asset.

The array is a complete list of the valid identifiers that can be used with exportSessionWithAsset with the specified asset.

Not all export presets are compatible with all assets. For example, a video-only asset is not compatible with an audio-only preset. This method returns only the identifiers for presets that are compatible with the given asset.

In order to ensure that the setup and running of an export operation will succeed using a given preset, you should not make significant changes to the asset (such as adding or deleting tracks) between retrieving compatible identifiers and performing the export operation.

If the asset's tracks are not currently loaded, they may be loaded by this method before any checks are performed.

4.1.33 exportSessionWithAsset(asset as AVAssetMBS, presetName as string) as AVAssetExportSessionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset export session configured with a specified asset and preset.
Notes: asset: The asset you want to export.
presetName: A string constant specifying the name of the preset template for the export.

Returns an asset export session initialized to export asset using preset presetName.

4.1.34 metadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the metadata to be written to the output file by the export session.

4.1.35 outputFileExtension as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Convenience method to query the preferred file extension for current output file type.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim p as string = AVAssetExportSessionMBS.AVAssetExportPresetAppleM4VCellular
dim e as new AVAssetExportSessionMBS(a, p)
dim filetypes() as string = e.supportedFileTypes
e.outputFileType = FileTypes(0)

// shows the output file extension
MsgBox "outputFileExtension: "+e.outputFileExtension
```

4.1.36 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the metadata to be written to the output file by the export session.
Example:

```
dim e as AVAssetExportSessionMBS // your export session
dim asset as AVAssetMBS // your asset

// query metadata
dim a() as AVMetadataItemMBS = asset.metadata

// make new
dim m as new AVMutableMetadataItemMBS

// add common key with author
m.keySpace = AVFoundationMBS.AVMetadataKeySpaceCommon
m.key = AVFoundationMBS.AVMetadataCommonKeyAuthor
m.Value = "Hello World"

// append to array and use as metadata:
a.Append m
e.setMetadata a
```

Notes: If the array is empty, any existing metadata in the exported asset will be translated as accurately as possible into the appropriate metadata key space for the output file and written to the output.

4.1.37 supportedFileTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The types of files the session can write. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim p as string = AVAssetExportSessionMBS.AVAssetExportPresetAppleM4VCellular
dim e as new AVAssetExportSessionMBS(a, p)
// shows supported file types
MsgBox join(e.supportedFileTypes, EndOfLine)
```

Notes: The types of files the session can write are determined by the asset and and export preset with which the session was initialized. If you need to determine the compatible file formats before initiating the export operation, use the determineCompatibleFileTypesWithCompletionHandler: method.

4.1.38 Properties

4.1.39 asset as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset with which the export session was initialized. (read-only)
Notes: (Read only property)

4.1.40 audioMix as AVAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether non-default audio mixing is enabled for export, and supplies the parameters for audio mixing.
Notes: (Read and Write property)

4.1.41 audioTimePitchAlgorithm as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates the processing algorithm used to manage audio pitch for scaled audio edits.
Notes: An exception (NSInvalidArgumentException) is raised if this property is set to a value other than

the constants defined in Time_Pitch_Algorithm_Settings.

The default value is AVAudioTimePitchAlgorithmSpectral.
Available in OS X v10.9 and later.
(Read and Write property)

4.1.42 canPerformMultiplePassesOverSourceMediaData as Boolean

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Determines whether the export session can perform multiple passes over the source media to achieve better results.
Notes: When the value for this property is true, the export session can produce higher quality results at the expense of longer export times. Setting this property to true may also require the export session to write temporary data to disk during the export. To control the location of temporary data, use the property directoryForTemporaryFiles.

The default value is false. Not all export session configurations can benefit from performing multiple passes over the source media. In these cases, setting this property to true has no effect.

This property cannot be set after the export has started.

Available in Mac OS X 10.10 and newer.
(Read and Write property)

4.1.43 customVideoCompositor as AVVideoCompositingMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates the custom video compositor instance used, if any. (read-only)
Notes: The custom video compositor instance that is used during image generation is accessible via this property after the value of videoComposition is set to an AVVideoComposition instance that specifies a custom video compositor class. Any additional communication between the application and that instance of the custom video compositor, if any is required for configuration or other purposes, can only occur once that has happened.

If the value of videoComposition is changed from an AVVideoComposition that specifies a custom video compositor class to another instance of AVVideoComposition that specifies the same custom video compositor class, the instance of the custom video compositor that was previously created will receive the renderContextChanged: message and remain in use for subsequent image generation.

This property is nil if there is no video compositor, or if the internal video compositor is in use.
Available in OS X v10.9 and later.
(Read only property)

4.1.44 directoryForTemporaryFiles as FolderItem

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Specifies a directory that is suitable for containing temporary files generated during the export process.
Notes: AVAssetExportSession may need to write temporary files when configured in certain ways, such as when canPerformMultiplePassesOverSourceMediaData is set to true. This property can be used to control where in the filesystem those temporary files are created. All temporary files will be deleted when the export is completed, is canceled, or fails.

When the value of this property is nil, the export session will choose a suitable location when writing temporary files. The default value is nil.

This property cannot be set after the export has started. The export will fail if the URL points to a location that is not a directory, does not exist, is not on the local file system, or if a file cannot be created in this directory (for example, due to insufficient permissions or sandboxing restrictions).

Available in Mac OS X 10.10 and newer.
(Read and Write property)

4.1.45 directoryForTemporaryFilesURL as String

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Specifies a directory that is suitable for containing temporary files generated during the export process.
Notes: AVAssetExportSession may need to write temporary files when configured in certain ways, such as when canPerformMultiplePassesOverSourceMediaData is set to true. This property can be used to control where in the filesystem those temporary files are created. All temporary files will be deleted when the export is completed, is canceled, or fails.

When the value of this property is empty, the export session will choose a suitable location when writing temporary files. The default value is nil.

This property cannot be set after the export has started. The export will fail if the URL points to a location that is not a directory, does not exist, is not on the local file system, or if a file cannot be created in this directory (for example, due to insufficient permissions or sandboxing restrictions).

Available in Mac OS X 10.10 and newer.
(Read and Write property)

4.1.46 error as NSErrorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Describes the error that occurred if the export status is AVAssetExportSessionStatusFailed or AVAssetExportSessionStatusCancelled. (read-only)
Notes: If there is no error to report, the value of this property is nil.
(Read only property)

4.1.47 estimatedOutputFileLength as Int64

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates the estimated size in bytes of the exported file. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.1.48 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.1.49 metadataItemFilter as AVMetadataItemFilterMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Specifies a filter object to be used during export to determine which metadata items should be transferred from the source asset.
Notes: If the value of this key is nil, no filter will be applied. This is the default.

The filter will not be applied to metadata set with via the metadata property. To apply the filter to metadata before it is set on the metadata property, see AVMetadataItem.

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

4.1.50 OutputFile as folderitem

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The file of the export session's output.
Notes: (Read and Write property)

4.1.51 outputFileType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The type of file to be written by the session.
Notes: The value is a UTI string corresponding to the file type to use when writing the asset. For a list of constants specifying UTIs for standard file types, see AV Foundation Constants Reference.
(Read and Write property)

4.1.52 outputURL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL of the export session's output.
Notes: (Read and Write property)

4.1.53 presetName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The name of the preset with which the session was initialized. (read-only)
Notes: (Read only property)

4.1.54 progress as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The progress of the export on a scale from 0 to 1. (read-only)
Notes: A value of 0 means the export has not yet begun, 1 means the export is complete.

(Read only property)

4.1.55 shouldOptimizeForNetworkUse as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the movie should be optimized for network use.
Notes: (Read and Write property)

4.1.56 status as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The status of the export session. (read-only)
Notes: (Read only property)

4.1.57 videoComposition as AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether video composition is enabled for export, and supplies the instructions for video composition.
Notes: (Read and Write property)

4.1.58 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range to be exported from the source.
Example:

```
dim a as AVAssetMBS // your asset to export
dim preset as string = AVAssetExportSessionMBS.AVAssetExportPresetPassthrough // your settings
dim e as new AVAssetExportSessionMBS(a, preset)

dim t as CMTimeMBS = CMTimeMBS.MakeWithSeconds(5) // start at 5
dim d as CMTimeMBS = CMTimeMBS.MakeWithSeconds(10) // duration 10
dim r as CMTimeRangeMBS = CMTimeRangeMBS.Make(t,d)

e.timeRange = r // or your time range!
e.shouldOptimizeForNetworkUse = true
```

```
dim filetypes() as string = e.supportedFileTypes
e.outputFileType = FileTypes(0)
e.OutputFile = SpecialFolder.Desktop.Child("test."+e.outputFileExtension)

e.exportAsynchronously(nil)

// keep e and check for events

// or instead of waiting for event, we simply wait for it to finish
do
app.YieldToNextThread
loop until e.status <>e.AVAssetExportSessionStatusExporting
```

Notes: The default time range of an export session is kCMTimeZero to kCMTimePositiveInfinity, meaning that (modulo a possible limit on file length) the full duration of the asset will be exported.
(Read and Write computed property)

4.1.59 Constants

Status Constants

Constant	Value	Description
AVAssetExportSessionStatusCancelled	5	Indicates that the export session was cancelled.
AVAssetExportSessionStatusCompleted	3	Indicates that the export session completed successfully.
AVAssetExportSessionStatusExporting	2	Indicates that the export session is in progress.
AVAssetExportSessionStatusFailed	4	Indicates that the export session failed.
AVAssetExportSessionStatusUnknown	0	Indicates that the status is unknown.
AVAssetExportSessionStatusWaiting	1	Indicates that the session is waiting to export more data.

4.2 class AVAssetImageGeneratorMBS

4.2.1 class AVAssetImageGeneratorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVAssetImageGenerator object provides thumbnail or preview images of assets independently of playback.
Notes: AVAssetImageGenerator uses the default enabled video track(s) to generate images. Generating a single image in isolation can require the decoding of a large number of video frames with complex interdependencies. If you require a series of images, you can achieve far greater efficiency using the asynchronous method, CGImageAtTime, which employs decoding efficiencies similar to those used during playback.

You create an asset generator using initWithAsset: or assetImageGeneratorWithAsset. These methods may succeed even if the asset possesses no visual tracks at the time of initialization. You can test whether an asset has any tracks with the visual characteristic using tracksWithMediaCharacteristic (AVAsset).

The actual time of a generated image is within the range [requestedTime-requestedTimeToleranceBefore, requestedTime+requestedTimeToleranceAfter] and may differ from the requested time for efficiency.

Assets that represent mutable compositions or mutable movies may gain visual tracks after initialization of an associated image generator.
Xojo Developer Magazine

- 12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.2.2 Methods

4.2.3 assetImageGeneratorWithAsset(asset as AVAssetMBS) as AVAssetImageGeneratorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an image generator for use with a specified asset.
Notes: asset: The asset from which images will be extracted.

Returns an image generator for use with asset.
This method may succeed even if the asset possesses no visual tracks at the time of initialization.

4.2.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.2.5 AVAssetImageGeneratorApertureModeCleanAperture as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify the aperture mode.
Notes: Both pixel aspect ratio and clean aperture will be applied..

4.2.6 AVAssetImageGeneratorApertureModeEncodedPixels as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify the aperture mode.
Notes: Neither pixel aspect ratio nor clean aperture will be applied.

4.2.7 AVAssetImageGeneratorApertureModeProductionAperture as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify the aperture mode.
Notes: Only pixel aspect ratio will be applied.

4.2.8 cancelAllCGImageGeneration

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels all pending image generation requests.
Notes: This method calls the AVFoundationMBS.generateCGImagesAsynchronouslyForTimesCompleted event with AVAssetImageGeneratorCancelled for each image time in every previous invocation of generateCGImagesAsynchronouslyForTimes for which images have not yet been supplied.

4.2.9 CGImageAtTime(time as CMTimeMBS, byref actualTime as CMTimeMBS, byref error as NSErrorMBS) as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a CGImage for the asset at or near a specified time.
Example:

```
// open asset
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)

// prepare generator
dim g as new AVAssetImageGeneratorMBS(a)
dim timeRequested as CMTimeMBS = CMTimeMBS.MakeWithSeconds(5, 30)
dim actualTime as CMTimeMBS
dim error as NSErrorMBS
// ask for an image
dim c as CGImageMBS = g.CGImageAtTime(timeRequested, actualTime, error)
// and show it
dim p as Picture = c.Picture
window1.backdrop = p
```

Notes: requestedTime: The time at which the image of the asset is to be created.
actualTime: Upon return, contains the time at which the image was actually generated.
Error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns a CGImageMBS for the asset at or near a specified time, or nil if the image could not be created.
This method returns the image synchronously.

4.2.10 Constructor(asset as AVAssetMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an image generator for use with a specified asset.
Notes: asset: The asset from which images will be extracted.
This method may succeed even if the asset possesses no visual tracks at the time of initialization.

4.2.11 generateCGImagesAsynchronouslyForTimes(times() as CMTimeMBS, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a series of CGImage objects for an asset at or near specified times.
Notes: requestedTimes: An array of CMTimeMBS, specifying the asset times at which an image is requested.

This method uses an efficient "batch mode" to get images in time order.

The client receives exactly one event call to AVFoundationMBS.generateCGImagesAsynchronouslyForTimesCompleted for each requested time in requestedTimes. Changes to the generator's properties (snap behavior, maximum size, and so on) do not affect pending asynchronous image generation requests.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.2.12 Properties

4.2.13 apertureMode as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies the aperture mode for the generated image.
Notes: (Read and Write property)

4.2.14 appliesPreferredTrackTransform as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies whether to apply the track matrix (or matrices) when extracting an image from the asset.
Notes: The default is false. AVAssetImageGenerator only supports rotation by 90, 180, or 270 degrees.
This property is ignored if you set a value for the videoComposition property.
(Read and Write property)

4.2.15 asset as AVAssetMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The asset with which the export session was initialized. (read-only)
Notes: Available in OS X v10.8 and later.
(Read only property)

4.2.16 customVideoCompositor as AVVideoCompositingMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates the custom video compositor instance used, if any. (read-only)
Notes: The custom video compositor instance that is used during image generation is accessible via this

property after the value of videoComposition is set to an AVVideoComposition instance that specifies a custom video compositor class. Any additional communication between the application and that instance of the custom video compositor, if any is required for configuration or other purposes, can only occur once that has happened.

If the value of videoComposition is changed from an AVVideoComposition that specifies a custom video compositor class to another instance of AVVideoComposition that specifies the same custom video compositor class, the instance of the custom video compositor that was previously created will receive the renderContextChanged message and remain in use for subsequent image generation.

This property is nil if there is no video compositor, or if the internal video compositor is in use.
Available in OS X v10.9 and later.
(Read only property)

4.2.17 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.2.18 videoComposition as AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video composition to use when extracting images from assets with multiple video tracks.
Notes: If no video composition is specified, only the first enabled video track will be used. If a video composition is specified, the appliesPreferredTrackTransform property is ignored.
(Read and Write property)

4.2.19 maximumSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies the maximum dimensions for generated image.
Notes: The default value is CGSizeZero, which specifies the asset's unscaled dimensions.

AVAssetImageGenerator scales images such that they fit within the defined bounding box. Images are never scaled up. The aspect ratio of the scaled image is defined by the apertureMode property.
(Read and Write computed property)

4.2.20 requestedTimeToleranceAfter as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The maximum length of time after a requested time for which an image may be generated.
Notes: The default value is kCMTimePositiveInfinity.

Set the values of requestedTimeToleranceBefore and requestedTimeToleranceAfter to kCMTimeZero to request frame-accurate image generation; this may incur additional decoding delay.
(Read and Write computed property)

4.2.21 requestedTimeToleranceBefore as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The maximum length of time before a requested time for which an image may be generated.
Notes: The default value is kCMTimePositiveInfinity.

Set the values of requestedTimeToleranceBefore and requestedTimeToleranceAfter to kCMTimeZero to request frame-accurate image generation; this may incur additional decoding delay.
(Read and Write computed property)

4.2.22 Constants

Result Constants

Constant	Value	Description
AVAssetImageGeneratorCancelled	2	Indicates that generation was cancelled.
AVAssetImageGeneratorFailed	1	Indicates that generation failed.
AVAssetImageGeneratorSucceeded	0	Indicates that generation succeeded.

4.3 class AVAssetMBS

4.3.1 class AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVAsset is an abstract class to represent timed audiovisual media such as videos and sounds.
Notes: Each asset contains a collection of tracks that are intended to be presented or processed together, each of a uniform media type, including but not limited to audio, video, text, closed captions, and subtitles.

An AVAsset object defines the collective properties of the tracks that comprise the asset. (You can access the instances of AVAssetTrack representing tracks of the collection, so you can examine each of these independently if you need to.) You often instantiate an asset using a concrete subclass of AVAsset; for example, you can initialize an instance of AVURLAsset using an URL that refers to an audiovisual media file, such as a QuickTime movie file or an MP3 files (amongst other types). You can also instantiate an asset using other concrete subclasses that extend the basic model for audiovisual media in useful ways, as AVComposition does for temporal editing. To assemble audiovisual constructs from one or more source assets, you can insert assets into instances of AVMutableComposition.

You often instantiate an asset using AVURLAsset—a concrete subclass of AVAsset—with URLs that refer to audiovisual media resources, such as streams (including HTTP live streams), QuickTime movie files, MP3 files, and files of other types. You can also instantiate an asset using other concrete subclasses that extend the basic model for audiovisual media in useful ways, as AVComposition does for temporal editing.

Properties of assets as a whole are defined by AVAsset. Additionally, references to instances of AVAssetTrack representing tracks of the collection can be obtained, so that each of these can be examined independently.

Because of the nature of timed audiovisual media, upon successful initialization of an asset some or all of the values for its keys may not be immediately available. The value of any key can be requested at any time, and asset will always return its value synchronously, although it may have to block the calling thread in order to do so. In order to avoid blocking, you can register your interest in particular keys and to become notified when their values become available. For further details, see AVAsynchronousKeyValueLoading.

To play an instance of AVAsset, initialize an instance of AVPlayerItem with it, use the player item tem to set up its presentation state (such as whether only a limited timeRange of the asset should be played, etc.), and provide the player item to an AVPlayer object according to whether the items is to be played by itself or together with a collection of other items.

You can insert AVAsset objects can also be inserted into an AVMutableComposition object in order to assemble audiovisual constructs from one or more source assets.

Subclassing Notes
It is not currently possible to subclass AVAsset to handle streaming protocols or file formats that are not

supported by the framework.
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 17.5
- MBS Xojo Plugins, version 17.5pr9
- MBS Xojo / Real Studio plug-ins in version 16.5
- MBS Xojo / Real Studio Plugins, version 16.5pr6
- Automated video editing with Xojo

Xojo Developer Magazine

- 16.1, page 9: News
- 12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.3.2 Methods

4.3.3 assetWithData(Data as MemoryBlock, Options as Dictionary = nil) as AVAssetMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in an data parameter.
Notes: data: An Memoryblock/String containing a movie header. We make a copy of that.
options: Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
You can use this method to operate on movie headers that are not stored in files; this might include movie headers on the pasteboard (which do not contain media data). In general you should avoid loading an entire movie file with its media data into an instance of Memoryblock/String! By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 4.3.4 assetWithData(Data as String, Options as Dictionary = nil) as AVAssetMBS 148

4.3.4 assetWithData(Data as String, Options as Dictionary = nil) as AVAssetMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in an data parameter.
Example:

```
// load movie in memory
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim b as BinaryStream = BinaryStream.Open(f)

// now open it from memory
dim s as string = b.Read(b.Length)
dim m as AVAssetMBS = AVAssetMBS.movieWithData(s)

MsgBox str(m.duration.Seconds)+" seconds."
```

Notes: data: An Memoryblock/String containing a movie header. We make a copy of that.
options: Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
You can use this method to operate on movie headers that are not stored in files; this might include movie headers on the pasteboard (which do not contain media data). In general you should avoid loading an entire movie file with its media data into an instance of Memoryblock/String! By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 4.3.3 assetWithData(Data as MemoryBlock, Options as Dictionary = nil) as AVAssetMBS 148

4.3.5 assetWithFile(file as folderitem) as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset for inspection of a media resource.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)

MsgBox str(a.duration.Seconds)+" seconds"
```

4.3.6 assetWithURL(URL as string) as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset for inspection of a media resource.

4.3.7 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.3.8 availableChapterLocales as NSLocaleMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The locales available for chapters in the asset. (read-only)

4.3.9 availableMediaCharacteristicsWithMediaSelectionOptions as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of media characteristics for which a media selection option is available. (read-only)
Notes: The value of this property is an array of strings, each string indicating a media characteristic for which a media selection option is available.

4.3.10 availableMetadataFormats as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of strings, each representing a metadata format that's available to the asset. (read-only)
Notes: Metadata formats may include ID3, iTunes metadata, and so on. For more details, see AVMetadataItem.

4.3.11 cancelLoading

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels the loading of all values for all observers.
Notes: Deallocation of an instance of the asset will implicitly invoke this method if any loading requests are still outstanding.

4.3.12 chapterMetadataGroupsBestMatchingPreferredLanguages as AVTimedMetadataGroupMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of chapters whose locale best matches the the list of preferred languages.
Notes: preferredLanguages: Optional, an array of strings, each of which contains a canonicalized IETF BCP 47 language identifier. The order of the identifiers in the array reflects the preferred language order, with the most preferred language being first in the array. Typically, you pass the user's preferred languages by retrieving this array from the preferredLanguages class method of NSLocale. If you don't pass an array, the plugin automatically passes NSLocaleMBS.preferredLanguages.

Returns an array of AVTimedMetadataGroup objects.

Each object in the returned array contains an AVMetadataItem object representing the chapter title. The time range property of the AVTimedMetadataGroup object is equal to the time range of the chapter title item.

All of the available chapter metadata is included in the metadata groups, including items with the common key AVMetadataCommonKeyArtwork, if such items are present. Items not carrying chapter titles are added to an existing AVTimedMetadataGroup object if the time range (timestamp and duration) of the metadata item and that of the metadata group overlaps. The locale of such items need not match the locale of the chapter titles.

You can use the metadataItemsFromArray:filteredAndSortedAccordingToPreferredLanguages: method to further filter the metadata items in each group. You can also filter the returned items based on locale using the metadataItemsFromArray:withLocale: method.

Special Considerations
Becomes callable without blocking when the data in the availableChapterLocales property is already loaded.
See also:

- 4.3.13 chapterMetadataGroupsBestMatchingPreferredLanguages(preferredLanguages() as string) as AVTimedMetadataGroupMBS() 151

4.3.13 chapterMetadataGroupsBestMatchingPreferredLanguages(preferredLanguages() as string) as AVTimedMetadataGroupMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of chapters whose locale best matches the the list of preferred languages.
Notes: preferredLanguages: Optional, an array of strings, each of which contains a canonicalized IETF BCP 47 language identifier. The order of the identifiers in the array reflects the preferred language order, with the most preferred language being first in the array. Typically, you pass the user's preferred languages

by retrieving this array from the preferredLanguages class method of NSLocale. If you don't pass an array, the plugin automatically passes NSLocaleMBS.preferredLanguages.

Returns an array of AVTimedMetadataGroup objects.

Each object in the returned array contains an AVMetadataItem object representing the chapter title. The time range property of the AVTimedMetadataGroup object is equal to the time range of the chapter title item.

All of the available chapter metadata is included in the metadata groups, including items with the common key AVMetadataCommonKeyArtwork, if such items are present. Items not carrying chapter titles are added to an existing AVTimedMetadataGroup object if the time range (timestamp and duration) of the metadata item and that of the metadata group overlaps. The locale of such items need not match the locale of the chapter titles.

You can use the metadataItemsFromArray:filteredAndSortedAccordingToPreferredLanguages: method to further filter the metadata items in each group. You can also filter the returned items based on locale using the metadataItemsFromArray:withLocale: method.

Special Considerations
Becomes callable without blocking when the data in the availableChapterLocales property is already loaded.
See also:

- 4.3.12 chapterMetadataGroupsBestMatchingPreferredLanguages as AVTimedMetadataGroupMBS() 151

4.3.14 chapterMetadataGroupsWithTitleLocale(locale as NSLocaleMBS, commonKeys() as string) as AVTimedMetadataGroupMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of chapters with a given title locale and containing specified keys.
Notes: locale: The locale of the metadata items carrying chapter titles to be returned (the method supports the IETF BCP 47 specification of locales).
commonKeys: An array of common keys of AVMetadataItem to include in the returned array. AVMetadataCommonKeyArtwork is the only supported key.

Returns an array of AVTimedMetadataGroup objects.

Each object in the returned array contains an AVMetadataItem object representing the chapter title, and the time range property of the AVTimedMetadataGroup object is equal to the time range of the chapter title item.

An AVMetadataItem with the specified common key is added to an existing AVTimedMetadataGroup object if the time range (timestamp and duration) of the metadata item and the metadata group overlap.

The locale of items not carrying chapter titles need not match the specified locale parameter. You can filter the returned items based on locale using metadataItemsFromArray.

4.3.15 commonMetadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of metadata items for each common metadata key for which a value is available. (read-only)
Notes: The value is an array of AVMetadataItem objects, one for each common metadata key for which a value is available. You can filter the array by locale using metadataItemsFromArray (AVMetadataItem) or by key using metadataItemsFromArray (AVMetadataItem).

4.3.16 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.3.17 copy as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.3.18 duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The duration of the asset. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)

MsgBox str(a.duration.Seconds)+" seconds"
```

Notes: If providesPreciseDurationAndTiming is false, a best-available estimate of the duration is returned. You can set the degree of precision required for timing-related properties at initialization time for assets initialized with URLs (see AVURLAssetPreferPreciseDurationAndTimingKey in AVURLAsset).

4.3.19 loadValuesAsynchronouslyForKeys(keys() as string, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the asset to load the values of any of the specified keys that are not already loaded.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetmbs.assetWithFile(f)

if a = nil then
MsgBox "failed to read file at all"
Return
end if

// request duration property
a.loadValuesAsynchronouslyForKeys array("duration")

dim e as NSErrorMBS
dim Status as Integer = a.statusOfValueForKey("duration", e)
while status <a.AVKeyValueStatusLoaded
// wait for loading in background
status = a.statusOfValueForKey("duration", e)
wend

// got it?
if status <>a.AVKeyValueStatusLoaded then
MsgBox "Failed to load duration."
if e <>nil then
MsgBox e.LocalizedDescription
end if

else
MsgBox str(a.duration.Seconds)+" seconds long."
end if
```

Notes: keys: An array containing the required keys.

Calls AVFoundationMBS.AssetLoadValuesAsynchronouslyForKeysFinished on completion.
The completion event will be invoked exactly once per invocation of this method:

- Synchronously if an I/O error or other format-related error occurs immediately.
- Asynchronously at a subsequent time if a loading error occurs at a later stage of processing, or if cancelLoading is invoked on an AVAsset instance.

The completion states of the keys you specify in keys are not necessarily the same—some may be loaded, and others may have failed. You must check the status of each key individually.

If you want to receive error reporting for loading that's still pending, you can call this method at any time—even after an asset has begun to load data for operations in progress or already completed. If a fatal error has already occurred, the completion event is invoked synchronously.

4.3.20 mediaSelectionGroupForMediaCharacteristic(mediaCharacteristic as string) as AVMediaSelectionGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an AVMediaSelectionGroup object that contains one or more options with the specified media characteristic.
Notes: mediaCharacteristic: A media characteristic for which you wish to obtain the available media selection options.
Only AVMediaCharacteristicAudible and AVMediaCharacteristicLegible are currently supported.
Pass AVMediaCharacteristicAudible to obtain the group of available options for audio media in various languages and for various purposes, such as descriptive audio.
Pass AVMediaCharacteristicLegible to obtain the group of available options for subtitles in various languages and for various purposes.

Returns an AVMediaSelectionGroup object that contains one or more options with the media characteristic specified by mediaCharacteristic, or nil if none could be found.

You can invoke this method without blocking when the key availableMediaCharacteristicsWithMediaSelectionOptions has been loaded.
You can filter the options in the returned media selection group according to playability, locale, and additional media characteristics can be accomplished using the filtering methods defined in the AVMediaSelectionGroup class.

4.3.21 metadata as AVMetadataItemMBS()

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Provides access to an array of AVMetadataItems for all metadata identifiers for which a value is available.
Example:

```
// load file and look for cover art

Dim file As FolderItem = SpecialFolder.Desktop.Child("test.m4a")
Dim a As AVURLAssetMBS = AVURLAssetMBS.URLAssetWithFile(file)

Dim iTunesMetadataKeyCoverArtString As String = AVFoundationMBS.AVMetadataiTunesMetadataKeyCoverArt
Dim iTunesMetadataKeyCoverArtNumber As Integer = OSTypeFromStringMBS(iTunesMetadataKeyCoverArtString)

Dim items() As AVMetadataItemMBS = a.metadata

For Each item As AVMetadataItemMBS In items

Dim key As Variant = item.key
Dim Val As Variant = item.value

If key.IntegerValue = iTunesMetadataKeyCoverArtNumber Or key.StringValue = iTunesMetadataKeyCoverArtString Then

Dim v As MemoryBlock = Val
window1.Backdrop = picture.FromData(v)
End If

Next
```

Notes: Available in Mac OS X 10.10.

4.3.22 metadataForFormat(Format as string) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of AVMetadataItem objects, one for each metadata item in the container of the specified format.
Notes: format: The metadata format for which you want items.

Returns an array of AVMetadataItem objects, one for each metadata item in the container of the specified format, or nil if there is no metadata of the specified format.

You can filter the array by locale using metadataItemsFromArray (AVMetadataItem) or by key using meta-

dataItemsFromArray (AVMetadataItem).
Becomes callable without blocking when availableMetadataFormats has been loaded.

4.3.23 naturalSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The encoded or authored size of the visual portion of the asset.

4.3.24 preferredTransform as CGAffineTransformMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The preferred transform to apply to the visual content of the asset for presentation or processing. (read-only)
Notes: The value is often, but not always, the identity transform.

4.3.25 readTimeCodeObjects as AVTimeCodeMBS()

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: Reads time codes from asset.
Notes: Raises exception if not possible.
Returns empty array if nothing found.

Provides details on time codes.

4.3.26 readTimeCodes as String()

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

Function: Reads time codes from asset.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim timecodes() as string = a.readTimeCodes

break // see in debugger
```

Notes: Raises exception if not possible.
Returns empty array if nothing found.

Format is 2 digit hours, double colon, 2 digit minutes, double colon, 2 digit seconds, double colon, 2 digit frames.
e.g. "01:02:03:04"

Please use readTimeCodeObjects for more options.

4.3.27 statusOfValueForKey(key as string, byref error as NSErrorMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Reports whether the value for a given key is immediately available without blocking. (required)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetmbs.assetWithFile(f)

if a = nil then
MsgBox "failed to read file at all"
Return
end if

// request duration property
a.loadValuesAsynchronouslyForKeys array("duration")

dim e as NSErrorMBS
dim Status as Integer = a.statusOfValueForKey("duration", e)
while status <a.AVKeyValueStatusLoaded
// wait for loading in background
status = a.statusOfValueForKey("duration", e)
wend

// got it?
if status <>a.AVKeyValueStatusLoaded then
MsgBox "Failed to load duration."
if e <>nil then
MsgBox e.LocalizedDescription
end if

else
MsgBox str(a.duration.Seconds)+" seconds long."
end if
```

Notes: key: The key whose status you want.
error: If the status of the value for the key is AVKeyValueStatusFailed, upon return contains an NSError object that describes the failure that occurred.

Returns the current loading status of the value for key. For possible values, see ”Protocol Methods.”

You use this method to determine the availability of the value for a key. This method does not cause an asset to load the value of a key that’s not yet available. To request values for keys that may not already be loaded without blocking, use loadValuesAsynchronouslyForKeys and wait for invocation of the completion handler to be informed of availability.

4.3.28 trackGroups as AVAssetTrackGroupMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array containing all of the track groups in the asset. (read-only)
Notes: This is an array of AVAssetTrackGroup instances, each representing a different grouping of tracks in the asset.
Available in OS X v10.9 and later.

4.3.29 tracks as AVAssetTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The tracks contained by the asset. (read-only)
Notes: Tracks are instances of AVAssetTrack.

4.3.30 tracksWithMediaCharacteristic(mediaCharacteristic as string) as AVAssetTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of AVAssetTrack objects of the asset that present media with a specified characteristic.
Notes: mediaCharacteristic: The media characteristic according to which receiver filters its asset tracks.
Return san array of AVAssetTrack objects that present media with mediaCharacteristic, or nil if no tracks with the specified characteristic are available.
You can call this method without blocking when tracks has been loaded.

4.3.31 tracksWithMediaType(mediaType as string) as AVAssetTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the asset tracks of the asset that present media of a specified type.
Notes: mediaType: The media type according to which the asset filters its tracks.
Media types are defined in AVAssetTrack.

Returns an array of AVAssetTrack objects of the asset that present media of mediaType.
You can call this method without blocking when tracks has been loaded.

4.3.32 trackWithTrackID(PersistentTrackID as Integer) as AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the track with a specified track ID.
Notes: trackID: The trackID of the requested asset track.

Returns the track with track ID trackID, or nil if no track with the specified ID is available.
You can call this method without blocking when tracks has been loaded.

4.3.33 unusedTrackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an ID that is currently unused by any of the tracks in the asset.

4.3.34 Properties

4.3.35 ClassName as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Returns the class name for the Objective-C class.
Notes: Sometimes useful for debugging.
(Read only property)

4.3.36 creationDate as AVMetadataItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the creation date of the asset. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)

dim creationDate as date = a.creationDate.dateValue
MsgBox creationDate.LongDate + " " + creationDate.LongTime
```

Notes: The value of this property may be nil.

If a creation date has been stored by the asset in a form that can be converted to an Date object, the dateValue property of the metadata item will provide an instance of NSDate. Otherwise the creation date is available only as a string value, using the AVMetadataItem stringValue method.
(Read only property)

4.3.37 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.3.38 hasProtectedContent as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset has protected content. (read-only)
Notes: (Read only property)

4.3.39 isComposable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset can be used within a segment of an AVCompositionTrack object. (read-only)
Notes: (Read only property)

4.3.40 isExportable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset can be exported using AVAssetExportSession. (read-only)
Notes: (Read only property)

4.3.41 isPlayable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset, or its URL, can be used to initialize an instance of AVPlayerItem. (read-only)
Notes: (Read only property)

4.3.42 isReadable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset's media data can be extracted using AVAssetReader. (read-only)
Notes: (Read only property)

4.3.43 lyrics as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The lyrics of the asset suitable for the current locale. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)

MsgBox "Lyrics: "+a.lyrics
```

Notes: (Read only property)

4.3.44 preferredRate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The natural rate at which the asset is to be played. (read-only)
Notes: This value is often, but not always, 1.0.
(Read only property)

4.3.45 preferredVolume as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The preferred volume at which the audible media of asset is to be played. (read-only)
Notes: This value is often, but not always, 1.0.
(Read only property)

4.3.46 providesPreciseDurationAndTiming as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the asset provides precise timing. (read-only)
Example:

```
dim options as new Dictionary
options.value(AVFoundationMBS.AVURLAssetPreferPreciseDurationAndTimingKey) = true
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVURLAssetMBS.URLAssetWithFile(f, options)

MsgBox str(a.duration.Seconds)+" seconds with precision: "+str(a.providesPreciseDurationAndTiming)
```

Notes: You can set the degree of precision required for timing-related properties at initialization time for assets initialized with URLs (see AVURLAssetPreferPreciseDurationAndTimingKey in AVURLAsset).
(Read only property)

4.3.47 referenceRestrictions as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The reference restrictions being used by the receiver. (read-only)
Notes: For AVURLAsset, this property reflects the value passed in for AVURLAssetReferenceRestrictionsKey, if any.

The default value for this property is AVAssetReferenceRestrictionForbidNone. See AVURLAssetReferenceRestrictionsKey for a full discussion of reference restrictions.

(Read only property)

4.3.48 Constants

Reference Restriction Modes

Constant	Value	Description
AVAssetReferenceRestrictionForbidAll	&hFFFF	Indicates that only references to media data sto file should be allowed.
AVAssetReferenceRestrictionForbidCrossSiteReference	4	Indicates that references from a remote asset a different site should not be followed.
AVAssetReferenceRestrictionForbidLocalReferenceToLocal	8	Indicates that references from a local asset to the asset's container file should not be followed
AVAssetReferenceRestrictionForbidLocalReferenceToRemote	2	Indicates that references from a local asset to be followed.
AVAssetReferenceRestrictionForbidNone	0	Indicates that all types of references should be
AVAssetReferenceRestrictionForbidRemoteReferenceToLocal	1	Indicates that references from a remote asset (URL) to local media data (for example, store followed.

Load Status Constants

Constant	Value	Description
AVKeyValueStatusCancelled	4	Indicates that the attempt to load the property was cancelled.
AVKeyValueStatusFailed	3	Indicates that the attempt to load the property failed.
AVKeyValueStatusLoaded	2	Indicates that the property is ready for use.
AVKeyValueStatusLoading	1	Indicates that the property is not fully loaded.
AVKeyValueStatusUnknown	0	Indicates that the property status is unknown.

4.4 class AVAssetReaderAudioMixOutputMBS

4.4.1 class AVAssetReaderAudioMixOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVAssetReaderAudioMixOutput is a concrete subclass of AVAssetReaderOutput that defines an interface for reading audio samples that result from mixing the audio from one or more tracks of an AVAssetReader object's asset.
Notes: You can read the audio data mixed from one or more asset tracks by adding an instance of AVAssetReaderAudioMixOutput to an asset reader using addOutput. The samples can be read in a default format or can be converted to a different format.
Subclass of the AVAssetReaderOutputMBS class.

4.4.2 Methods

4.4.3 assetReaderAudioMixOutputWithAudioTracks(audioTracks() as AVAssetTrackMBS, outputSettings as dictionary) as AVAssetReaderAudioMixOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an instance of AVAssetReaderAudioMixOutput for reading mixed audio from the specified audio tracks, with optional audio settings.
Notes: audioTracks: An array of AVAssetTrack objects from which the created object should read sample buffers to be mixed.
Each track must be one of the tracks owned by the target AVAssetReader object's asset and must be of media type AVMediaTypeAudio.
audioSettings: The audio settings to be used for audio output; the dictionary must contain values for keys in AVAudioSettings.h (linear PCM only). Pass nil if you want to receive decoded samples in a convenient uncompressed format, with properties determined according to the properties of the specified audio tracks.

Returns an instance of AVAssetReaderAudioMixOutput for reading mixed audio from audioTracks, with audio settings specified by audioSettings.
Initialization will fail if audioSettings cannot be used with audioTracks.

4.4.4 audioSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The audio settings used for audio output. (read-only)
Notes: The dictionary must contain values for keys in AVAudioSettings.h (linear PCM only).
nil indicates that the samples will be returned in the default format.

4.4.5 audioTracks as AVAssetTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The tracks from which the receiver reads mixed audio. (read-only)
Notes: The value is an array of AVAssetTrack objects owned by the target AVAssetReader object's asset.

4.4.6 Constructor(audioTracks() as AVAssetTrackMBS, outputSettings as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an instance of AVAssetReaderAudioMixOutput for reading mixed audio from the specified audio tracks, with optional audio settings.
Notes: audioTracks: An array of AVAssetTrack objects from which the created object should read sample buffers to be mixed.
Each track must be one of the tracks owned by the target AVAssetReader object's asset and must be of media type AVMediaTypeAudio.
audioSettings: The audio settings to be used for audio output; the dictionary must contain values for keys in AVAudioSettings.h (linear PCM only).
Pass nil if you want to receive decoded samples in a convenient uncompressed format, with properties determined according to the properties of the specified audio tracks.

Returns an instance of AVAssetReaderAudioMixOutput initialized for reading mixed audio from audioTracks, with audio settings specified by audioSettings.

Initialization will fail if audioSettings cannot be used with audioTracks. So please check handle property.

4.4.7 Properties

4.4.8 audioMix as AVAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The output's audio mix.
Notes: You use the audio mix to specify how the volume of audio samples read from each source track will change over the timeline of the source asset.
(Read and Write property)

4.4.9 audioTimePitchAlgorithm as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The processing algorithm used to manage audio pitch for scaled audio edits.
Notes: The supported constants are defined in Time Pitch Algorithm Settings.
An NSInvalidArgumentException will be raised if this property is set to a value other than the defined constants.
Available in OS X v10.9 and later.
(Read and Write property)

4.5 class AVAssetReaderMBS

4.5.1 class AVAssetReaderMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVAssetReader object to obtain media data of an asset, whether the asset is file-based or represents an assemblage of media data from multiple sources (as with an AVComposition object).
Notes: AVAssetReader lets you:

Read raw un-decoded media samples directly from storage, obtain samples decoded into renderable forms.
Mix multiple audio tracks of the asset and compose multiple video tracks (by using AVAssetReaderAudioMixOutput and AVAssetReaderVideoCompositionOutput).
AVAssetReader's pipelines are multithreaded internally. After you initiate reading with initWithAsset:error:, a reader loads and processes a reasonable amount of sample data ahead of use so that retrieval operations such as copyNextSampleBuffer (AVAssetReaderOutput) can have very low latency. Note, however, that AVAssetReader is not intended for use with real-time sources, and its performance is not guaranteed for real-time operations.

4.5.2 Methods

4.5.3 addOutput(output as AVAssetReaderOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds a given output to the receiver.
Notes: output: The reader output to add.

Outputs are created with a reference to one or more AVAssetTrack objects. Adding an output to an asset reader indicates to the reader that it should source from those tracks. The tracks must be owned by the asset returned by the reader's asset property.
You cannot add an output after reading has started.

4.5.4 asset as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset with which the receiver was initialized. (read-only)
Notes: Concrete instances of AVAssetReader with specific AVAssetTrack instances must obtain those tracks from the asset returned by this property.

4.5.5 assetReaderWithAsset(item as AVAssetMBS, byref error as NSErrorMBS) as AVAssetReaderMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset reader for reading media data from a specified asset.
Notes: asset: The asset from which media data is to be read.
Error: If initialization of the reader fails, upon return contains an error that describes the problem.

Returns an asset reader, initialized for reading media data from asset.

4.5.6 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.5.7 canAddOutput(output as AVAssetReaderOutputMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given output can be added to the receiver.
Notes: output: The reader output to be tested.

Returns true if output can be added to the receiver, otherwise false.
You cannot add an output that reads from a track of an asset other than the asset used to initialize the receiver.

4.5.8 cancelReading

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels any background work and prevents the receiver's outputs from reading more samples.
Notes: If you want to stop reading samples from the receiver before reaching the end of its time range, you should call this method to stop any background read ahead operations that the may have been in progress.

4.5.9 Constructor(item as AVAssetMBS, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset reader for reading media data from a specified asset.
Notes: asset: The asset from which media data is to be read.
Error: If initialization of the reader fails, upon return contains an error that describes the problem.

4.5.10 error as NSErrorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Describes the error that occurred if the status is AVAssetReaderStatusFailed. (read-only)
Notes: This property is thread safe.

The value of this property describes what caused the reader to no longer be able to read its asset. If the reader's status is not AVAssetReaderStatusFailed, the value of this property is nil.

4.5.11 outputs as AVAssetReaderOutputMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The outputs from which clients of reader can read media data. (read-only)
Notes: The array contains concrete instances of AVAssetReaderOutput associated with the reader.

4.5.12 startReading

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prepares the receiver for obtaining sample buffers from the asset.
Notes: Returns true if the reader is able to start reading, otherwise false.

This method validates the entire collection of settings for outputs for tracks, for audio mixdown, and for video composition and initiates reading of all outputs.

status signals the terminal state of the asset reader, and if a failure occurs, error describes the failure.

4.5.13 status as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The status of the reading of sample buffers from the asset. (read-only)
Notes: This property is thread safe. For possible values, see ”Reader Status Constants.”

The value of this property indicates whether reading is in progress, has completed successfully, has been canceled, or has failed. You should check the value of this property copyNextSampleBuffer (AVAssetReaderOutput) returns nil to determine why no more samples could be read.

4.5.14 Properties

4.5.15 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.5.16 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range of the asset that should be read.
Notes: The intersection of the value of this property and CMTimeRangeMake(kCMTimeZero, asset.duration) determines the time range of the asset from which media data will be read.

The default value is CMTimeRangeMake(kCMTimeZero, kCMTimePositiveInfinity). You cannot change the value of this property after reading has started.
(Read and Write computed property)

4.5.17 Constants

Status Values

Constant	Value	Description
AVAssetReaderStatusCancelled	4	Indicates that reading was cancelled using cancelReading.
AVAssetReaderStatusCompleted	2	Indicates that the reader has provided all available sample buffers to all of its outputs.
AVAssetReaderStatusFailed	3	Indicates that reading failed.
AVAssetReaderStatusReading	1	Indicates that the reader is ready to provide more sample buffers to its outputs.
AVAssetReaderStatusUnknown	0	Indicates that startReading has not yet been invoked.

4.6 class AVAssetReaderOutputMBS

4.6.1 class AVAssetReaderOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVAssetReaderOutput is an abstract class that defines an interface for reading a single collection of samples of a common media type from an AVAssetReader object.
Notes: There are several subclasses of AVAssetReaderOutput for specific tasks, such as AVAssetReaderTrackOutput or AVAssetReaderVideoCompositionOutput.

You can read the media data of an asset by adding one or more concrete instances of AVAssetReaderOutput to an AVAssetReader object using addOutput.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.6.2 Methods

4.6.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.6.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.6.5 NextSampleBuffer as CMSampleBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Synchronously copies the next sample buffer for the output.
Notes: The output sample buffer, or nil if there are no more sample buffers available for the output within the time range specified by the asset reader's timeRange property.

If this method returns nil, you should check the value of the associated AVAssetReader object's status property to determine why no more samples could be read.

4.6.6 Properties

4.6.7 alwaysCopiesSampleData as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the data in buffers gets copied before being vended.
Notes: When the value of this property is true, the output always vends a buffer with copied data—you can freely modify data in such buffers.

When the value of this property is false, the buffers vended may not be copied—such buffers may still be referenced by other entities. The result of modifying a buffer whose data hasn't been copied is undefined.
Requesting buffers whose data hasn't been copied when possible can lead to performance improvements.
The default value of this property is true.
Available in OS X v10.8 and later.
(Read and Write property)

4.6.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.6.9 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A string representing the media type of the track (or tracks) represented by the output. (read-only)
Notes: The value of this property is one of the media type strings defined in AVMediaFormat.h.
(Read only property)

4.6.10 supportsRandomAccess as Boolean

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Indicates whether the asset reader output supports reconfiguration of the time ranges to read.
Notes: When the value of this property is true, the time ranges read by the asset reader output can be reconfigured during reading using the -resetForReadingTimeRanges method. This also prevents the attached AVAssetReader from progressing to AVAssetReaderStatusCompleted until markConfigurationAsFinal has

been invoked.

The default value is false, which means that the asset reader output may not be reconfigured once reading has begin. When the value of this property is false, AVAssetReader may be able to read media data more efficiently, particularly when multiple asset reader outputs are attached.

This property may not be set after startReading has been called on the attached asset reader.
(Read and Write property)

4.7 class AVAssetReaderOutputMetadataAdaptorMBS

4.7.1 class AVAssetReaderOutputMetadataAdaptorMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Creates a new timed metadata group adaptor for retrieving timed metadata group objects from an asset reader output.
Notes: assetReaderOutput: An instance of AVAssetReaderTrackOutput that vends sample buffers containing metadata, e.g. an AVAssetReaderTrackOutput object initialized with a track of media type AVMediaTypeMetadata and nil outputSettings.

Returns an instance of AVAssetReaderOutputMetadataAdaptor

It is an error to create a timed metadata group adaptor with an asset reader output that does not vend metadata. It is also an error to create a timed metadata group adaptor with an asset reader output whose asset reader has already started reading, or an asset reader output that already has been used to initialize another timed metadata group adaptor.

Clients should not mix calls to AVAssetReaderTrackOutput.NextSampleBuffer and AVAssetReaderOutputMetadataAdaptor.nextTimedMetadataGroup. Once an AVAssetReaderTrackOutput instance has been used to initialize an AVAssetReaderOutputMetadataAdaptor, calling NextSampleBuffer on that instance will result in an exception being thrown.

Available on Mac OS X 10.10 or newer.

4.7.2 Methods

4.7.3 assetReaderOutputMetadataAdaptorWithAssetReaderTrackOutput(trackOutput as AVAssetReaderTrackOutputMBS) as AVAssetReaderOutputMetadataAdaptorMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Creates a new timed metadata group adaptor for retrieving timed metadata group objects from an asset reader output.
Notes: Parameter: an instance of AVAssetReaderTrackOutput that vends sample buffers containing metadata, e.g. an AVAssetReaderTrackOutput object initialized with a track of media type AVMediaTypeMetadata and nil outputSettings.

Returns an instance of AVAssetReaderOutputMetadataAdaptor

It is an error to create a timed metadata group adaptor with an asset reader output that does not vend metadata. It is also an error to create a timed metadata group adaptor with an asset reader output whose asset reader has already started reading, or an asset reader output that already has been used to initialize another timed metadata group adaptor.

Clients should not mix calls to AVAssetReaderTrackOutput.NextSampleBuffer and AVAssetReaderOutputMetadataAdaptor.nextTimedMetadataGroup. Once an AVAssetReaderTrackOutput instance has been used to initialize an AVAssetReaderOutputMetadataAdaptor, calling NextSampleBuffer on that instance will result in an exception being thrown.

4.7.4 available as boolean

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Should be true on Mac OS X 10.10 and newer.

4.7.5 Constructor(trackOutput as AVAssetReaderTrackOutputMBS)

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Creates a new timed metadata group adaptor for retrieving timed metadata group objects from an asset reader output.
Notes: assetReaderOutput: An instance of AVAssetReaderTrackOutput that vends sample buffers containing metadata, e.g. an AVAssetReaderTrackOutput object initialized with a track of media type AVMediaTypeMetadata and nil outputSettings.

Returns an instance of AVAssetReaderTrackOutputTimedMetadataGroupAdaptor

It is an error to create a timed metadata group adaptor with an asset reader output that does not vend metadata. It is also an error to create a timed metadata group adaptor with an asset reader output whose asset reader has already started reading, or an asset reader output that already has been used to initialize another timed metadata group adaptor.

Clients should not mix calls to AVAssetReaderTrackOutput.NextSampleBuffer and AVAssetReaderOutputMetadataAdaptor.nextTimedMetadataGroup. Once an AVAssetReaderTrackOutput instance has been used to initialize an AVAssetReaderOutputMetadataAdaptor, calling NextSampleBuffer on that instance will result in an exception being thrown.

4.7.6 nextTimedMetadataGroup as AVTimedMetadataGroupMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Returns the next timed metadata group for the asset reader output, synchronously.
Notes: Returns an instance of AVTimedMetadataGroup, representing the next logical segment of metadata coming from the source asset reader output.

This method will return nil when all timed metadata groups have been read from the asset reader output, or if there is an error that prevents the timed metadata group adaptor from reading more timed metadata groups. When this method returns nil, clients should check the value of the associated AVAssetReader's status property to determine why no more samples could be read.

Before calling this method, you must ensure that the output which underlies the receiver is attached to an AVAssetReader via a prior call to addOutput and that startReading has been called on the asset reader.

4.7.7 Properties

4.7.8 assetReaderTrackOutput as AVAssetReaderTrackOutputMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: The asset reader track output from which the receiver pulls timed metadata groups.
Notes: (Read only property)

4.7.9 Handle as Integer

Plugin Version: 14.4, Platform: macOS, Targets: All.

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

4.8 class AVAssetReaderSampleReferenceOutputMBS

4.8.1 class AVAssetReaderSampleReferenceOutputMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: AVAssetReaderSampleReferenceOutput is a concrete subclass of AVAssetReaderOutput that defines an interface for reading sample references from a single AVAssetTrack of an AVAssetReader's AVAsset.
Notes: Clients can extract information about the location (file URL and offset) of samples in a track by adding an instance of AVAssetReaderSampleReferenceOutput to an AVAssetReader using the AVAssetReader.addOutput method. No actual sample data can be extracted using this class. The location of the sample data is described by the kCMSampleBufferAttachmentKey_SampleReferenceURL and kCMSampleBufferAttachmentKey_SampleReferenceByteOffset attachments on the extracted sample buffers. More information about sample buffers describing sample references can be found in the CMSampleBuffer documentation.

Sample buffers extracted using this class can also be appended to an AVAssetWriterInput to create movie tracks that are not self-contained and reference data in the original file instead. Currently, only instances of AVAssetWriter configured to write files of type AVFileTypeQuickTimeMovie can be used to write tracks that are not self-contained.

Since no sample data is ever returned by instances of AVAssetReaderSampleReferenceOutput, the value of the alwaysCopiesSampleData property is ignored.

Available on Mac OS X 10.10 or newer.
Subclass of the AVAssetReaderOutputMBS class.

4.8.2 Methods

4.8.3 assetReaderSampleReferenceOutputWithTrack(track as AVAssetTrackMBS) as AVAssetReaderSampleReferenceOutputMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Returns an instance of AVAssetReaderSampleReferenceOutput for supplying sample references.
Notes: Track: The AVAssetTrack for which the resulting AVAssetReaderSampleReferenceOutput should provide sample references.

Returns an instance of AVAssetReaderSampleReferenceOutput.
The track must be one of the tracks contained by the target AVAssetReader's asset.

4.8.4 Constructor(Track as AVAssetTrackMBS)

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Returns an instance of AVAssetReaderSampleReferenceOutput for supplying sample references.
Notes: Track: The AVAssetTrack for which the resulting AVAssetReaderSampleReferenceOutput should provide sample references.

The track must be one of the tracks contained by the target AVAssetReader's asset.

4.8.5 Properties

4.8.6 track as AVAssetTrackMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: The track from which the receiver extracts sample references.
Notes: The value of this property is an AVAssetTrack owned by the target AVAssetReader's asset.
(Read only property)

4.9 class AVAssetReaderTrackOutputMBS

4.9.1 class AVAssetReaderTrackOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVAssetReaderTrackOutput defines an interface for reading media data from a single AVAssetTrack object of an asset reader's asset.
Notes: You can read the media data of an asset track by adding an instance of AVAssetReaderTrackOutput to an asset reader using the AVAssetReader's addOutput method. The samples in the track can be read in read in the format in which they are stored in the asset, or can be converted to a different format.
Subclass of the AVAssetReaderOutputMBS class.
Blog Entries

- MBS Xojo Plugins, version 23.3pr5

4.9.2 Methods

4.9.3 assetReaderTrackOutputWithTrack(track as AVAssetTrackMBS, outputSettings as dictionary) as AVAssetReaderTrackOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset reader wrapping a specified track, with optional output settings.
Notes: track: The track from which the reader should source sample buffers.
outputSettings: A dictionary of output settings to be used for sample output. Pass nil to receive samples as stored in the track.
You use keys from one of AVAudioSettings.h, AVVideoSettings.h, or <CoreVideo/CVPixelBuffer.h>, depending on the media type and the output format you want.
Initialization fails if the output settings cannot be used with the specified track.
Those keys are available in AVFoundationMBS class.

4.9.4 Constructor(track as AVAssetTrackMBS, outputSettings as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset reader to wrap a specified track, with optional output settings.
Notes: track: The track from which the reader should source sample buffers.
outputSettings: A dictionary of output settings to be used for sample output. Pass nil to receive samples as stored in the track.
You use keys from one of AVAudioSettings.h, AVVideoSettings.h, or <CoreVideo/CVPixelBuffer.h>, depending on the media type and the output format you want.
Initialization fails if the output settings cannot be used with the specified track.
Those keys are available in AVFoundationMBS class.

4.9.5 outputSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The output settings used by the output. (read-only)
Notes: The value is a dictionary that contains values for keys from either AVAudioSettings.h (linear PCM only) for audio tracks or <CoreVideo/CVPixelBuffer.h>for video tracks. A value of nil indicates that the output will return samples in their original format as stored in the target track.
Those keys are available in AVFoundationMBS class.

4.9.6 track as AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The track from which the receiver reads sample buffers. (read-only)

4.9.7 Properties

4.9.8 audioTimePitchAlgorithm as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The processing algorithm used to manage audio pitch for scaled audio edits.
Notes: The supported constants are defined in Time Pitch Algorithm Settings.
An NSInvalidArgumentException will be raised if this property is set to a value other than the defined constants.
Available in OS X v10.9 and later.
(Read and Write computed property)

4.10 class AVAssetReaderVideoCompositionOutputMBS

4.10.1 class AVAssetReaderVideoCompositionOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVAssetReaderVideoCompositionOutput is a subclass of AVAssetReaderOutput you use to read video frames that have been composited together from the frames in one or more tracks of an AVAssetReader object's asset.
Notes: You can read the video frames composited from one or more asset tracks by adding an instance of AVAssetReaderVideoCompositionOutput to an AVAssetReader object using the addOutput method.
Subclass of the AVAssetReaderOutputMBS class.

4.10.2 Methods

4.10.3 assetReaderVideoCompositionOutputWithVideoTracks(videoTracks() as AVAssetTrackMBS, videoSettings as dictionary) as AVAssetReaderVideoCompositionOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an instance of AVAssetReaderVideoCompositionOutput for reading composited video from the specified video tracks, using optional video settings.
Notes: videoTracks: An array of AVAssetTrack objects from which the created object should read video frames for compositing.
It is an error to include tracks of media types other than AVMediaTypeVideo.
videoSettings: A dictionary of video settings to be used for sample output, or nil if you want to receive decoded samples in a convenient uncompressed format, with properties determined according to the properties of the specified video tracks.
You use keys from <CoreVideo/CVPixelBuffer.h>, depending on the output format you want.
Initialization will fail if the video settings cannot be used with the specified video tracks.

Return san instance of AVAssetReaderVideoCompositionOutput wrapping videoTracks, using the settings specified by videoSettings, or nil if initialization failed.

4.10.4 Constructor(videoTracks() as AVAssetTrackMBS, videoSettings as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an instance of AVAssetReaderVideoCompositionOutput for reading composited video from the specified video tracks, using optional video settings.
Notes: videoTracks: An array of AVAssetTrack objects from which the created object should read video

frames for compositing.
It is an error to include tracks of media types other than AVMediaTypeVideo.
videoSettings: A dictionary of video settings to be used for sample output, or nil if you want to receive decoded samples in a convenient uncompressed format, with properties determined according to the properties of the specified video tracks.
You use keys from <CoreVideo/CVPixelBuffer.h>, depending on the output format you want.
Initialization will fail if the video settings cannot be used with the specified video tracks.
In that case handle property will be 0.

4.10.5 customVideoCompositor as AVVideoCompositingMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the custom video compositor instance used by the receiver, if any. (read-only)
Notes: The custom video compositor instance that is used during image generation is accessible via this property after the value of videoComposition is set to an AVVideoComposition instance that specifies a custom video compositor class. Any additional communication between the application and that instance of the custom video compositor, if any is required for configuration or other purposes, can only occur once that has happened.

If the value of videoComposition is changed from an AVVideoComposition that specifies a custom video compositor class to another instance of AVVideoComposition that specifies the same custom video compositor class, the instance of the custom video compositor that was previously created will receive the renderContextChanged message and remain in use for subsequent image generation.

This property is nil if there is no video compositor, or if the internal video compositor is in use.
Available in OS X v10.9 and later.

4.10.6 videoSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video settings used by the output. (read-only)
Notes: A value of nil indicates that the receiver will return video frames in a convenient uncompressed format, with properties determined according to the properties of the receiver's video tracks.

The dictionary's keys are from <CoreVideo/CVPixelBuffer.h>.

4.10.7 videoTracks as AVAssetTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The tracks from which the output reads composited video. (read-only)
Notes: The array contains AVAssetTrack objects owned by the target asset reader's asset.

4.10.8 Properties

4.10.9 videoComposition as AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video composition to use for the output.
Notes: The value is an AVVideoComposition object that can be used to specify the visual arrangement of video frames read from each source track over the timeline of the source asset.

See AVVideoComposition for options for configuring a video composition.
(Read and Write computed property)

4.11 class AVAssetResourceLoaderMBS

4.11.1 class AVAssetResourceLoaderMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The asset resource loader class.
Notes: An AVAssetResourceLoader object mediates resource requests from an AVURLAsset object with a delegate object that you provide. When a request arrives, the resource loader asks your delegate if it is able to handle the request and reports the results back to the asset.

You do not create resource loader objects yourself. Instead, you retrieve a resource loader from the resourceLoader property of an AVURLAsset object and use it to assign your custom delegate object.

The delegate you associate with this object must adopt the AVAssetResourceLoaderDelegate protocol. For more information, see AVAssetResourceLoaderDelegate Protocol Reference.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.11.2 Methods

4.11.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 and newer.

4.11.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.11.5 Properties

4.11.6 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.12 class AVAssetResourceLoadingContentInformationRequestMBS

4.12.1 class AVAssetResourceLoadingContentInformationRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVAssetResourceLoadingContentInformationRequest class represents a query for essential information about a resource referenced by an asset resource loading request.
Notes: The event AVFoundationMBS.resourceLoaderShouldWaitForLoadingOfRequestedResource is invoked and accepts responsibility for loading the resource it must check whether the contentInformation-Request property of the AVAssetResourceLoadingRequest is not nil. Whenever the value is not nil, the request includes a query for the information that AVAssetResourceLoadingContentInformationRequest encapsulates. In response to such queries, the resource loading delegate should set the values of the content information request's properties appropriately before invoking the AVAssetResourceLoadingRequest method finishLoading.

When finishLoading is invoked, the values of the properties of its contentInformationRequest property will, in part, determine how the requested resource is processed. For example, if the requested resource's URL is the URL of an AVURLAsset and contentType is set by the resource loading delegate to a value that the underlying media system doesn't recognize as a supported media file type, operations on the AVURLAsset, such as playback, are likely to fail.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.12.2 Methods

4.12.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 and newer.

4.12.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.12.5 Properties

4.12.6 ByteRangeAccessSupported as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether random access to arbitrary ranges of bytes of the resource is supported.
Notes: Before finishing loading an AVAssetResourceLoadingRequest instance, if its contentInformationRequest property is not nil, set the value of this property to true if it supports random access to arbitrary ranges of bytes of the resource.

If this property is not true for resources that must be loaded incrementally, loading of the resource may fail. Such resources include anything that contains media data.

If byte range access is supported portions of the resource can be requested more than once.
Available in OS X v10.9 and later.
(Read and Write property)

4.12.7 contentLength as Int64

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The length, in bytes, of the requested resource.
Notes: Before finishing loading an AVAssetResourceLoadingRequest instance, if its contentInformationRequest property is not nil, set the value of the contentLength property to the number of bytes contained by the requested resource.
Available in OS X v10.9 and later.
(Read and Write property)

4.12.8 contentType as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The UTI that specifies the type of data contained by the requested resource.
Notes: Before finishing loading an AVAssetResourceLoadingRequest instance, if its contentInformationRequest property is not nil, set the value of this property to a UTI indicating the type of data contained by the requested resource.
Available in OS X v10.9 and later.
(Read and Write property)

4.12.9 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.13 class AVAssetResourceLoadingDataRequestMBS

4.13.1 class AVAssetResourceLoadingDataRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Use the AVAssetResourceLoadingDataRequest class to request data from a resource referenced by an AVAssetResourceLoadingRequest instance.
Notes: The AVAssetResourceLoaderDelegate uses the AVAssetResourceLoadingDataRequest class to do the actual data reading, and its methods will be invoked, as necessary, to acquire data for the AVAssetResourceLoadingRequest instance.

When the resource loading delegate, which implements the AVAssetResourceLoaderDelegate protocol, receives an instance of AVAssetResourceLoadingRequest as the second parameter of the delegate's resourceLoader:shouldWaitForLoadingOfRequestedResource: method, it has the option of accepting responsibility for loading the referenced resource. If it accepts that responsibility, by returning YES, it must check whether the dataRequest property of the AVAssetResourceLoadingRequest instance is not nil. If it is not nil, the resource loading delegate is informed of the range of bytes within the resource that are required by the underlying media system. In response, the data is provided by one or more invocations of respondWithData: as required to provide the requested data. The data can be provided in increments determined by the resource loading delegate according to convenience or efficiency.

When the AVAssetResourceLoadingRequest method finishLoading is invoked, the data request is considered fully satisfied. If the entire range of bytes requested has not yet been provided, the underlying media system assumes that the resource's length is limited to the provided content.

Available in OS X v10.9 and later.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.13.2 Methods

4.13.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.

4.13.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.13.5 respondWithData(data as MemoryBlock)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Provides data to the loading request.
Notes: data: An instance of NSData containing some or all of the requested bytes.

This method may be invoked multiple times on the same instance of AVAssetResourceLoadingDataRequest to provide the full range of requested data incrementally. Upon each invocation, the value of the currentOffset property is updated to match the amount of data provided.

Available in OS X v10.9 and later.

4.13.6 Properties

4.13.7 currentOffset as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The position within the resource of the next byte. (read-only)
Notes: When incrementally loading data you should begin loading at this offset, returning the data by invoking the respondWithData: method. Bytes previous to this value have already been provided.

Available in OS X v10.9 and later.
(Read only property)

4.13.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.13.9 requestedLength as Int64

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The length, in bytes, of the data requested. (read-only)

Notes: If the content length of the resource is unknown, the sum of the requestedLength and requestedOffset properties may be greater than the actual content length. When this situation occurs, an application must attempt to provide as much of the requested data beginning at the requestedOffset property as the resource contains. The application must then invoke either the AVAssetResourceLoadingRequest instance's finishLoading method upon success, or the finishLoadingWithError: method if an error is encountered during the loading.
Available in OS X v10.9 and later.
(Read only property)

4.13.10 requestedOffset as Int64

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The position within the resource of the first byte requested. (read-only)
Notes: When all of the requested bytes that can be provided have been loaded—including the possible contentInformationRequest data in the AVAssetResourceLoadingRequest instance that contains the receiver—the delegate should respond by invoking finishLoading.

If the requestedOffset value is beyond the content length of the resource, the AVAssetResourceLoadingRequest instance is sent a finishLoading message without any prior invocations of respondWithData.
Available in OS X v10.9 and later.
(Read only property)

4.14 class AVAssetResourceLoadingRequestMBS

4.14.1 class AVAssetResourceLoadingRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An AVAssetResourceLoadingRequest object encapsulates information about a resource request issued from a resource loader object.
Notes: When an AVURLAsset object needs help loading a resource, it asks its AVAssetResourceLoader object to assist. The resource loader encapsulates the request information by creating an instance of this object, which it then hands to its delegate object for processing. The delegate uses the information in this object to perform the request and report on the success or failure of the operation.
Available on Mac OS X 10.9 and later.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.14.2 Methods

4.14.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.

4.14.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.14.5 finishLoading

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Causes the receiver to treat the processing of the request as complete.
Notes: If a dataRequest is present and the resource does not contain the full extent of the data that has been requested according to the values of the requestedOffset and requestedLength properties of the request, invoke finishLoading after providing as much of the requested data as the resource contains.

Available in OS X v10.9 and later.
See also:

- 4.14.6 finishLoading(error as NSErrorMBS) 194

4.14.6 finishLoading(error as NSErrorMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Causes the receiver to handle the failure to load a resource for which a resource loader's delegate took responsibility.
Notes: error: An error object indicating the reason for the failure.
When a resource loader's delegate takes responsibility for loading a resource, it calls this method when a failure occurred when loading the resource. This method marks the loading request as finished and notifies the resource loader object that the resource could not be loaded.
Available in OS X v10.9 and later.
See also:

- 4.14.5 finishLoading 193

4.14.7 streamingContentKeyRequestDataForApp(appIdentifier as Memoryblock, contentIdentifier as Memoryblock, options as Dictionary, byref error as NSErrorMBS) as Memoryblock

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Obtains key request data for a specific combination of application and content.
Notes: appIdentifier: An opaque content identifier for the application. The value of this identifier depends on the particular system used to provide the decryption key.
contentIdentifier: An opaque identifier for the content. The value of this identifier depends on the particular system used to provide the decryption key.
options: Additional information necessary to obtain the key, or nil if no additional information is required.
Error: If an error occurs while obtaining the streaming content key, the pointer is set to an appropriate error object on output.

Returns the key request data that must be transmitted to the key vendor to obtain the content key.
Available in OS X v10.9 and later.

4.14.8 Properties

4.14.9 contentInformationRequest as AVAssetResourceLoadingContentInformationRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Information about the requested resource. (read-only)
Notes: An instance of AVAssetResourceLoadingContentInformationRequest that you populate with information about the resource. The value of this property is nil if no such information is being requested.
Available in OS X v10.9 and later.

(Read only property)

4.14.10 dataRequest as AVAssetResourceLoadingDataRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The range of requested resource data. (read-only)
Notes: An instance of AVAssetResourceLoadingDataRequest that indicates the range of resource data that's being requested. The value of this property is nil if no data is being requested.
Available in OS X v10.9 and later.
(Read only property)

4.14.11 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.14.12 isCancelled as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the request has been cancelled. (read-only)
Notes: True when the resource loader cancels the loading of a request, just prior to sending the message resourceLoaderDidCancelLoadingRequest to the delegate.
(Read only property)

4.14.13 isFinished as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether loading of the resource has finished. (read-only)
Notes: The value of this property is NO initially. The value changes to YES when the delegate object handling the request calls the finishLoadingWithResponse or finishLoadingWithError method.
Available in OS X v10.9 and later.
(Read only property)

4.14.14 redirect as Variant

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An URL request instance if the loading request was redirected.
Notes: Value is NSURLRequestMBS.
Set this property to an instance of NSURLRequest indicating a redirection of the loading request to another URL.
If no redirection is needed, the value of this property must be nil, which is the default.
Available in OS X v10.9 and later.
(Read only property)

4.14.15 request as Variant

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The URL request object for the resource. (read-only)
Notes: Value is NSURLRequestMBS.
Use the value in this property to identify the requested resource and to formulate an appropriate response object.
Available in OS X v10.9 and later.
(Read only property)

4.14.16 response as Variant

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The URL response for the loading request.
Notes: Value is NSURLResponseMBS.
The value of this property to an instance of NSURLResponse, indicating a response to the loading request.
If no response is needed, the value of this property is nil.
Available in OS X v10.9 and later
(Read only property)

4.15 class AVAssetTrackGroupMBS

4.15.1 class AVAssetTrackGroupMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVAssetTrackGroup class encapsulates a single group of related tracks in an asset.
Notes: Instances of AVAssetTrackGroup describe a single group of related alternate tracks, only one of which should be played at a time. Groups of alternate tracks typically contain variations of the same content, such as subtitles in multiple translations.

Clients can inspect the track groups contained in an asset (AVAsset) instance by loading and obtaining the value of the asset's trackGroups property.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.15.2 Methods

4.15.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 and newer.

4.15.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.15.5 copy as AVAssetTrackGroupMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates a copy of the track group.

4.15.6 trackIDs as Integer()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The IDs of the tracks in the group. (read-only)
Notes: The value of this property is an array of NSNumber instances that are interpreted as CMPersistentTrackID values, one for each track in the group.
Available in OS X v10.9 and later.

4.15.7 Properties

4.15.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.16 class AVAssetTrackMBS

4.16.1 class AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVAssetTrack object provides provides the track-level inspection interface for all assets.
Notes: AVAssetTrack adopts the AVAsynchronousKeyValueLoading protocol. You should use methods in the protocol to access a track's properties without blocking the current thread. To cancel load requests for all keys of AVAssetTrack you must message the parent AVAsset object (for example, [track.asset cancelLoading]).
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Automated video editing with Xojo

- MBS Xojo / Real Studio Plugins, version 15.4pr1

4.16.2 Methods

4.16.3 associatedTracksOfType(trackAssociationType as string) as AVAssetTrackMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array containing other tracks associated with the track using the specified association type.
Notes: trackAssociationType: The association type for which associated tracks are requested.

Returns an array of AVAssetTrack objects associated with the track by trackAssociationType. May be empty if no tracks are associated through the specified association type.

You can call this method without blocking after availableTrackAssociationTypes has been loaded.
Available in OS X v10.9 and later.

4.16.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.16.5 availableMetadataFormats as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array containing the metadata formats available for the track. (read-only)
Notes: The array contains String objects, one for each metadata format that's available for the track (such as QuickTime user data). For possible values, see AVMetadataItem.

4.16.6 availableTrackAssociationTypes as string()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array of association types used to associate other tracks with the track. (read-only)
Notes: The array contains strings, one for each type of association for which the track has associated tracks.
For possible values, see "Track Association Types" in Apple's documentation.
Available in OS X v10.9 and later.

4.16.7 commonMetadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVMetadataItem objects for each common metadata key for which a value is available. (read-only)

4.16.8 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.16.9 copy as AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.16.10 formatDescriptions as CMFormatDescriptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The formats of media samples referenced by the track. (read-only)
Notes: The array contains CMFormatDescriptions, each of which indicates the format of media samples referenced by the track. A track that presents uniform media (for example, encoded according to the same encoding settings) will provide an array with a count of 1.

4.16.11 hasMediaCharacteristic(mediaCharacteristic as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the track references media with the specified media characteristic.
Notes: mediaCharacteristic: The media characteristic of interest.
For possible values, see "Media Characteristics" in AV Foundation Constants Reference, for example AVMediaCharacteristicVisual, AVMediaCharacteristicAudible, or AVMediaCharacteristicLegible.

Returns true if the track references media with the specified characteristic, otherwise false.

4.16.12 loadValuesAsynchronouslyForKeys(keys() as string, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the asset to load the values of any of the specified keys that are not already loaded.
Notes: keys: An array containing the required keys.

Calls AVFoundationMBS.AssetTrackLoadValuesAsynchronouslyForKeysFinished on completion.
The completion event will be invoked exactly once per invocation of this method:

- Synchronously if an I/O error or other format-related error occurs immediately.
- Asynchronously at a subsequent time if a loading error occurs at a later stage of processing, or if cancelLoading is invoked on an AVAsset instance.

The completion states of the keys you specify in keys are not necessarily the same—some may be loaded, and others may have failed. You must check the status of each key individually.

If you want to receive error reporting for loading that's still pending, you can call this method at any time—even after an asset has begun to load data for operations in progress or already completed. If a fatal error has already occurred, the completion event is invoked synchronously.

4.16.13 metadataForFormat(Format as string) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of metadata items, one for each metadata item in the container of the specified format.
Notes: format: The metadata format for which items are requested.

Returns an array of AVMetadataItem objects, one for each metadata item in the container of the format specified by format, or empty array if there is no metadata of the specified format.

You can call this method without blocking after availableMetadataFormats has been loaded.

4.16.14 preferredTransform as CGAffineTransformMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The transform specified in the track's storage container as the preferred transformation of the visual media data for display purposes. (read-only)
Notes: The value of this property is often, but not always, CGAffineTransformIdentity.

4.16.15 samplePresentationTimeForTrackTime(trackTime as CMTimeMBS) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Maps the specified track time through the appropriate time mapping and returns the resulting sample presentation time.
Notes: trackTime: The track time for which a sample presentation time is requested.
Returns the sample presentation time corresponding to trackTime; the value will be invalid if trackTime is out of range.

4.16.16 segmentForTrackTime(trackTime as CMTimeMBS) as AVAssetTrackSegmentMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The track segment that corresponds to the specified track time.
Notes: trackTime: The track time for which you want the segment.
Returns the track segment from the segments array that corresponds to trackTime, or nil if trackTime is out of range.

4.16.17 segments as AVAssetTrackSegmentMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time mappings from the track's media samples to the timeline of the track. (read-only)
Notes: The array contains instances of AVAssetTrackSegment.

Empty edits (that is, time ranges for which no media data is available to be presented) have source.start and source.duration equal to kCMTimeInvalid.

4.16.18 statusOfValueForKey(key as string, byref error as NSErrorMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Reports whether the value for a given key is immediately available without blocking. (required)
Notes: key: The key whose status you want.
error: If the status of the value for the key is AVKeyValueStatusFailed, upon return contains an NSError object that describes the failure that occurred.

Returns the current loading status of the value for key. For possible values, see "Protocol Methods."

You use this method to determine the availability of the value for a key. This method does not cause an asset to load the value of a key that's not yet available. To request values for keys that may not already be loaded without blocking, use loadValuesAsynchronouslyForKeys and wait for invocation of the completion handler to be informed of availability.

4.16.19 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range of the track within the overall timeline of the asset. (read-only)
Notes: A track with CMTimeCompare(timeRange.start, kCMTimeZero) == 1 will initially present an empty time range.

4.16.20 trackSamples(formatOptions as Dictionary) as MemoryBlock

Plugin Version: 15.4, Platform: macOS, Targets: All.

Function: Queries the track Samples
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim tracks() as AVAssetTrackMBS = a.tracksWithMediaType(AVFoundationMBS.AVMediaTypeAudio)
dim track as AVAssetTrackMBS = tracks(0)

dim d as new Dictionary
dim n as Integer = OSTypeFromStringMBS(AVFoundationMBS.kAudioFormatLinearPCM)
d.Value(AVFoundationMBS.AVFormatIDKey) = n
d.Value(AVFoundationMBS.AVLinearPCMBitDepthKey) = 16
d.Value(AVFoundationMBS.AVLinearPCMIsBigEndianKey) = false
d.Value(AVFoundationMBS.AVLinearPCMIsFloatKey) = false
d.Value(AVFoundationMBS.AVLinearPCMIsNonInterleaved) = false

dim SampleData as MemoryBlock = track.trackSamples(d)

window1.Backdrop = RenderSamplesMBS(SampleData, SampleData.Size / 2, 1, window1.Width, window1.Height,
1, &c88B5C4, &c274C5A, &c203F4E, 15)
```

Notes: Returns nil in case of errors.
Track should be an audio or muxed track.

4.16.21 Properties

4.16.22 asset as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset of which the track is a part. (read-only)
Notes: (Read only property)

4.16.23 estimatedDataRate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The estimated data rate of the media data referenced by the track, in bits per second. (read-only)
Notes: (Read only property)

4.16.24 extendedLanguageTag as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The language tag associated with the track, as an RFC 4646 language tag. (read-only)
Notes: The value may be nil if no language tag is indicated.
(Read only property)

4.16.25 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.16.26 isEnabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the track is enabled according to state stored in its container or construct. (read-only)
Notes: (Read only property)

4.16.27 isPlayable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the track is playable in the current environment. (read-only)
Notes: If the value of this property is true, an AVPlayerItemTrack of an AVPlayerItem initialized with the the track's asset can be enabled for playback.
Available in OS X v10.8 and later.
(Read only property)

4.16.28 isSelfContained as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the track references sample data only within its storage container. (read-only)
Notes: The value is true if the track references sample data only within its storage container, otherwise it is false.
(Read only property)

4.16.29 languageCode as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The language associated with the track, as an ISO 639-2/T language code. (read-only)
Notes: The value may be "" if no language is indicated.
(Read only property)

4.16.30 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The media type for the track. (read-only)
Notes: For possible values, see "Media Types" in AV Foundation Constants Reference.
(Read only property)

4.16.31 naturalSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The natural dimensions of the media data referenced by the track. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

dim videotracks() as AVMovieTrackMBS = m.movieTracksWithMediaCharacteristic(AVFoundationMBS.AVMediaCharacteristicFrameBased)

for each videotrack as AVMovieTrackMBS in videotracks
dim s as CGSizeMBS = videotrack.naturalSize
MsgBox str(s.Width)+" x "+ str(s.Height)
next
```

Notes: (Read only property)

4.16.32 naturalTimeScale as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A timescale in which time values for the track can be operated upon without extraneous numerical conversion. (read-only)

Notes: (Read only property)

4.16.33 nominalFrameRate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The frame rate of the track, in frames per second. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp4")
dim a as AVAssetMBS = AVAssetMBS.assetWithFile(f)
dim tracks() as AVAssetTrackMBS = a.tracksWithMediaCharacteristic(AVFoundationMBS.AVMediaCharacteristicFrameBased)

for each t as AVAssetTrackMBS in tracks
MsgBox str(t.nominalFrameRate)
next
```

Notes: (Read only property)

4.16.34 preferredVolume as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The volume specified in the track's storage container as the preferred volume of the audible media data. (read-only)
Notes: (Read only property)

4.16.35 totalSampleDataLength as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The total number of bytes of sample data required by the track. (read-only)
Notes: (Read only property)

4.16.36 trackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The persistent unique identifier for this track of the asset. (read-only)

Notes: (Read only property)

4.16.37 Constants

Load Status Constants

Constant	Value	Description
AVKeyValueStatusCancelled	4	Indicates that the attempt to load the property was cancelled.
AVKeyValueStatusFailed	3	Indicates that the attempt to load the property failed.
AVKeyValueStatusLoaded	2	Indicates that the property is ready for use.
AVKeyValueStatusLoading	1	Indicates that the property is not fully loaded.
AVKeyValueStatusUnknown	0	Indicates that the property status is unknown.

4.17 class AVAssetTrackSegmentMBS

4.17.1 class AVAssetTrackSegmentMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVAssetTrackSegment object represents a segment of an AVAssetTrack object, comprising of a time mapping from the source to the asset track timeline.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.17.2 Methods

4.17.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.17.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.17.5 isEmpty as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the segment is an empty segment (read-only)
Notes: True if the segment is empty, otherwise false.

4.17.6 timeMapping as CMTimeMappingMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range of the track of the container file of the media presented by the segment. (read-only)

4.17.7 Properties

4.17.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.18 class AVAssetWriterInputGroupMBS

4.18.1 class AVAssetWriterInputGroupMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVAssetWriterInputGroup class associates tracks corresponding to inputs with each other in a mutually exclusive relationship.
Notes: This class is used to associate tracks corresponding to multiple AVAssetWriterInput instances as mutually exclusive to each other for playback or other processing.

For example, if you are creating an asset with multiple audio tracks using different spoken languages–and only one track should be played at a time–group the inputs corresponding to those tracks into a single instance of AVAssetWriterInputGroup and add the group to the AVAssetWriter instance using the AVAssetWriter method addInputGroup:. If the output format supports mutually exclusive relationships among tracks, the AVAssetWriter marks the tracks as mutually exclusive to each other.

Because the AVAssetWriterInputGroup class is a subclass of the AVMediaSelectionGroup class, clients can examine the media selection options that are available on the output asset before the asset is written. The best results for examining the options of the AVAssetWriterInputGroup are obtained after associating the AVAssetWriterInput instances of the AVAssetTrack objects as appropriate using the AVAssetWriterInput method addTrackAssociationWithTrackOfInput and by initializing each AVAssetWriterInput with a source format hint, where appropriate.
Subclass of the AVMediaSelectionGroupMBS class.

4.18.2 Methods

4.18.3 assetWriterInputGroupWithInputs(inputs() as AVAssetWriterInputMBS, defaultInput as AVAssetWriterInputMBS) as AVAssetWriterInputGroupMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates and initializes an instance of an asset writer input group.
Notes: inputs: An array of AVAssetWriterInput instances to be grouped together.
defaultInput: The instance of AVAssetWriterInput to designate as the default input for the group.

Returns an initialized instance of AVAssetWriterInputGroup.
When the input group is added to an AVAssetWriter instance using the addInputGroup: method, the AVAssetWriterInput property marksOutputTrackAsEnabled is automatically set to true for defaultInput and false for all of the other inputs in the group.
Available in OS X v10.9 and later.

4.18.4 Constructor(inputs() as AVAssetWriterInputMBS, defaultInput as AVAssetWriterInputMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Initializes an instance of an asset writer input group.
Notes: inputs: An array of AVAssetWriterInput instances to be grouped together.
defaultInput: The instance of AVAssetWriterInput to designate as the default input for the group.

Returns an initialized instance of AVAssetWriterInputGroup.

When the input group is added to an AVAssetWriter instance using the addInputGroup method, the AVAssetWriterInput property marksOutputTrackAsEnabled is automatically set to true for defaultInput and false for all of the other inputs in the group.
Available in OS X v10.9 and later.

4.18.5 inputs as AVAssetWriterInputMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The the inputs array. (read-only)
Notes: Available in OS X v10.9 and later.

4.18.6 Properties

4.18.7 defaultInput as AVAssetWriterInputMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The the default input object. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.19 class AVAssetWriterInputMBS

4.19.1 class AVAssetWriterInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVAssetWriterInput to append media samples packaged as CMSampleBuffer objects, or collections of metadata, to a single track of the output file of an AVAssetWriter object.
Notes: When there are multiple inputs, AVAssetWriter tries to write media data in an ideal interleaving pattern for efficiency in storage and playback. Each of its inputs signals its readiness to receive media data for writing according to that pattern via the value of readyForMoreMediaData. If readyForMoreMediaData is true, an input can accept additional media data while maintaining appropriate interleaving. If media data is appended to an input after readyForMoreMediaData becomes false, AVAssetWriter may need to write media data to its output without regard for ideal interleaving.

You can only append media data to an input while its readyForMoreMediaData property is true.

- If you're writing media data from a non-real-time source, such as an instance of AVAssetReader, you should hold off on generating or obtaining more media data to append to an input when the value of readyForMoreMediaData is false. To help with control of the supply of non-real-time media data, you can use requestMediaDataWhenReadyOnQueue:usingBlock: to specify a block that the input should invoke whenever it's ready for input to be appended.
- If you're writing media data from a real-time source, you should set the input's expectsMediaDataInRealTime property to true to ensure that the value of readyForMoreMediaData is calculated appropriately. When expectsMediaDataInRealTime is true, readyForMoreMediaData will become false only when the input cannot process media samples as quickly as they are being provided by the client. If readyForMoreMediaData becomes false for a real-time source, the client may need to drop samples or consider reducing the data rate of appended samples.

The value of readyForMoreMediaData will often change from false to true asynchronously, as previously-supplied media data is processed and written to the output. It is possible for all of an asset writer's inputs temporarily to return false for readyForMoreMediaData.
Xojo Developer Magazine

- 12.2, page 47: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.19.2 Methods

4.19.3 addTrackAssociationWithTrackOfInput(input as AVAssetWriterInputMBS, trackAssociationType as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Associates the track corresponding to the specified input with the track corresponding with the receiver.
Notes: input: The instance of AVAssetWriterInput with a corresponding track to associate with track corresponding with the receiver.
trackAssociationType: The type of track association to add. Common track association types, such as AVTrackAssociationTypeTimecode are defined in Track Association Types.

If the type of association requires tracks of specific media types that don't match the media types of the inputs, or if the output file type does not support track associations, an NSInvalidArgumentException is raised.

Note: Track associations cannot be added after writing on the receiver's AVAssetWriter instance has started.
Available in OS X v10.9 and later.

4.19.4 appendSampleBuffer(sampleBuffer as CMSampleBufferMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Appends samples to the receiver.
Notes: sampleBuffer: The CMSampleBuffer to be appended.

Returns true if sampleBuffer as appended successfully, otherwise false.

The timing information in the sample buffer, considered relative to the time passed to the asset writer's startSessionAtSourceTime: will be used to determine the timing of those samples in the output file.

Do not modify sampleBuffer or its contents after you have passed it to this method.

4.19.5 assetWriterInputWithMediaType(MediaType as string, outputSettings as dictionary = nil) as AVAssetWriterInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new input of the specified media type to receive sample buffers for writing to the output file.
Notes: mediaType: The media type of samples that will be accepted by the input.
Media types are defined in AVMediaFormat.h.
outputSettings: The settings used for encoding the media appended to the output. Pass nil to specify that appended samples should not be re-encoded.
Audio output settings keys are defined in AVAudioSettings.h. Video output settings keys are defined in AVVideoSettings.h. Video output settings with keys from <CoreVideo/CVPixelBuffer.h>are not currently supported.

Returns a new input of the specified media type to receive sample buffers for writing to the output file.

Each new input accepts data for a new track of the asset writer's output file. You add an input to an asset writer using the AVAssetWriter method addInput:.

Passing nil for outputSettings instructs the input to pass through appended samples, doing no processing before they are written to the output file. This is useful if, for example, you are appending buffers that are already in a desirable compressed format. However, passthrough is currently supported only when writing to QuickTime Movie files (i.e. the AVAssetWriter was initialized with AVFileTypeQuickTimeMovie). For other file types, you must specify non-nil output settings.
See also:

- 4.19.6 assetWriterInputWithMediaType(MediaType as string, outputSettings as dictionary, sourceFormatHint as CMFormatDescriptionMBS) as AVAssetWriterInputMBS 215

4.19.6 assetWriterInputWithMediaType(MediaType as string, outputSettings as dictionary, sourceFormatHint as CMFormatDescriptionMBS) as AVAssetWriterInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new input of the specified media type to receive sample buffers for writing to the output file.
See also:

- 4.19.5 assetWriterInputWithMediaType(MediaType as string, outputSettings as dictionary = nil) as AVAssetWriterInputMBS 214

4.19.7 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.19.8 canAddTrackAssociationWithTrackOfInput(input as AVAssetWriterInputMBS, trackAssociationType as string) as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether an association between the tracks corresponding to a pair of inputs is valid.
Notes: input: The instance of AVAssetWriterInput with a corresponding track to associate with track cor-

responding with the receiver.
trackAssociationType: The type of track association to test. Common track association types, such as AVTrackAssociationTypeTimecode are defined in Track Association Types.

Returns true if the track association can be added; otherwise false.

If the type of association requires tracks of specific media types that don't match the media types of the inputs, or if the output file type does not support track associations, returns NO.
Available in OS X v10.9 and later.

4.19.9 Constructor(MediaType as string, outputSettings as dictionary = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initialized a new input of the specified media type to receive sample buffers for writing to the output file.
Notes: mediaType: The media type of samples that will be accepted by the input.
Media types are defined in AVMediaFormat.h.
outputSettings: The settings used for encoding the media appended to the output. Pass nil to specify that appended samples should not be re-encoded.
Audio output settings keys are defined in AVAudioSettings.h. Video output settings keys are defined in AVVideoSettings.h. Video output settings with keys from <CoreVideo/CVPixelBuffer.h>are not currently supported.

Returns an input of the specified media type initialized to receive sample buffers for writing to the output file.

Each new input accepts data for a new track of the asset writer's output file. You add an input to an asset writer using the AVAssetWriter method addInput:.

Passing nil for outputSettings instructs the input to pass through appended samples, doing no processing before they are written to the output file. This is useful if, for example, you are appending buffers that are already in a desirable compressed format. However, passthrough is currently supported only when writing to QuickTime Movie files (i.e. the AVAssetWriter was initialized with AVFileTypeQuickTimeMovie). For other file types, you must specify non-nil output settings.
See also:

- 4.19.10 Constructor(MediaType as string, outputSettings as dictionary, sourceFormatHint as CMFormatDescriptionMBS) 217

4.19.10 Constructor(MediaType as string, outputSettings as dictionary, sourceFormatHint as CMFormatDescriptionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initialized a new input of the specified media type to receive sample buffers for writing to the output file.
See also:

- 4.19.9 Constructor(MediaType as string, outputSettings as dictionary = nil) 216

4.19.11 markAsFinished

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the writer that no more buffers will be appended to this input.
Notes: If you are monitoring each input's expectsMediaDataInRealTime value to keep the output file well interleaved, it is important to call this method when you have finished adding buffers to a track. This is necessary to prevent other inputs from stalling, as they may otherwise wait forever for that input's media data, attempting to complete the ideal interleaving pattern.

4.19.12 metadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The collection of track-level metadata for association with the asset and for carriage in the output file.
Notes: The array contains AVMetadataItem objects representing the collection of track-level metadata to be written in the output file.
See also setMetadata method.

4.19.13 outputSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The settings used for encoding the media appended to the output. (read-only)
Notes: A value of nil specifies that appended samples should not be re-encoded.

4.19.14 requestMediaDataWhenReadyOnQueue(assetWriterInput as AVAssetWriterInputMBS, AssetReaderOutput as AVAssetReaderOutputMBS, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the receiver to invoke an event repeatedly, at its convenience, in order to gather media data for writing to the output.
Notes: This is a special version of that method where the plugin does the required work.
The plugin provides the block to do the work to copy samples from input to output.
You receive AVFoundationMBS.requestMediaDataWhenReadyOnQueueProgress events and on the end an AVFoundationMBS.requestMediaDataWhenReadyOnQueueFinished event, when assetReaderOutput.NextSampleBuffer does not provide the next buffer.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.19.15 requestMediaDataWhenReadyOnQueue(tag as Variant = nil) 218

4.19.15 requestMediaDataWhenReadyOnQueue(tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the receiver to invoke an event repeatedly, at its convenience, in order to gather media data for writing to the output.
Notes: Calls later AVFoundationMBS.requestMediaDataWhenReadyOnQueueCompleted event.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.19.14 requestMediaDataWhenReadyOnQueue(assetWriterInput as AVAssetWriterInputMBS, AssetReaderOutput as AVAssetReaderOutputMBS, tag as Variant = nil) 218

4.19.16 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the collection of track-level metadata for association with the asset and for carriage in the output file.
Notes: The array contains AVMetadataItem objects representing the collection of track-level metadata to be written in the output file.

You cannot set this property after writing on the receiver's asset writer has started.

4.19.17 sourceFormatHint as CMFormatDescriptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A hint about the format of buffers that will be appended.
Notes: An AVAssetWriterInput object may be able to use this hint to fill in missing output settings or perform more upfront validation. To guarantee successful file writing, if you set this property you should ensure that subsequently-appended buffers are of the specified format. An NSInvalidArgumentException will be thrown if the media type of the format description does not match the media type of the writer input.

4.19.18 Properties

4.19.19 expectsMediaDataInRealTime as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the input should tailor its processing of media data for real-time sources.
Notes: If you are appending media data to an input from a real-time source, such as an AVCaptureOutput, you should set expectsMediaDataInRealTime to true. This will ensure that readyForMoreMediaData is calculated appropriately for real-time usage.
(Read and Write property)

4.19.20 extendedLanguageTag as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Language tag to associate with the track corresponding to the receiver.
Notes: The value is specified as an RFC 4646 language tag; can be nil in which case no tag is written to the track.

Extended language tags are normally set only when an ISO 639-2/T language code by itself is ambiguous, as in cases in which media data should be distinguished not only by language but also by the regional dialect in use or the writing system employed.
Available in OS X v10.9 and later.
(Read and Write property)

4.19.21 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.19.22 isReadyForMoreMediaData as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the readiness of the input to accept more media data. (read-only)
Notes: This property is observable using key-value observing (see Key-Value Observing Programming Guide). Observers should not assume that they will be notified of changes on a specific thread.
(Read only property)

4.19.23 languageCode as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Language to associate with the track corresponding to the receiver.
Notes: The value is specified as an ISO 639-2/T language code; can be nil in which case no language code is written to the track..
Available in OS X v10.9 and later.
(Read and Write property)

4.19.24 marksOutputTrackAsEnabled as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns whether the receiver's track is enabled.
Notes: For file types that support enabled and disabled tracks, such as QuickTime Movie files, specifies whether the track corresponding to the receiver should be enabled by default for playback and processing.

The default value is true.
Available in OS X v10.9 and later.
(Read and Write property)

4.19.25 mediaTimeScale as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies the media time scale to be used.
Notes: For file types that support media time scales, such as QuickTime Movie files, specifies the media time scale to be used.
The default value is 0, which indicates that you should choose a convenient value, if applicable.
You cannot set this property after writing has started.
(Read and Write property)

4.19.26 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The media type of the samples that can be appended to the input. (read-only)
Notes: The value of this property is one of the media type strings defined in AVMediaFormat.
(Read only property)

4.19.27 naturalSize as CGSizeMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Size specified in the output file as the natural dimensions of the visual media data for display.
Notes: If the default value, (0,0), is specified, the naturalSize of the track corresponding to the receiver is set according to dimensions indicated by the format descriptions that are ultimately written to the output track.

Note: This property cannot be set after writing on the receiver's AVAssetWriter has started.
Available in OS X v10.9 and later.
(Read only property)

4.19.28 preferredVolume as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Preferred volume level to be stored in the output file.
Notes: The value for this property should typically be in the range of 0.0 to 1.0.
The default value is 1.0, which is equivalent to a ”normal” volume level.

Note: This property cannot be set after writing on the receiver's AVAssetWriter has started.
Available in OS X v10.9 and later.
(Read and Write property)

4.19.29 transform as CGAffineTransformMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The transform specified in the output file as the preferred transformation of the visual media data for display purposes.
Notes: If no value is specified, the identity transform is used.
(Read and Write computed property)

4.20 class AVAssetWriterInputPixelBufferAdaptorMBS

4.20.1 class AVAssetWriterInputPixelBufferAdaptorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVAssetWriterInputPixelBufferAdaptor to append video samples packaged as CVPixelBuffer objects to a single AVAssetWriterInput object.
Notes: Instances of AVAssetWriterInputPixelBufferAdaptor provide a CVPixelBufferPool that you can use to allocate pixel buffers for writing to the output file. Using the provided pixel buffer pool for buffer allocation is typically more efficient than appending pixel buffers allocated using a separate pool.
Blog Entries

- JPEGs to Movie
- MBS Xojo / Real Studio Plugins, version 15.2pr9
- MBS Xojo / Real Studio Plugins, version 15.2pr2

Xojo Developer Magazine

- 12.2, page 47: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.20.2 Methods

4.20.3 appendPicture(pic as picture, presentationTime as CMTimeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Appends a picture to the receiver.
Notes: Internally creates a copy of the picture to create a PixelBuffer and calls appendPixelBuffer.
Works in Cocoa and Carbon. Using appendPixelBuffer directly with PixelBuffer objects may be more efficient depending on what your app does.

4.20.4 appendPixelBuffer(pixelBuffer as CVPixelBufferMBS, presentationTime as CMTimeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Appends a pixel buffer to the receiver.
Notes: pixelBuffer: The CVPixelBuffer to be appended.
presentationTime: The presentation time for the pixel buffer to be appended. This time will be considered relative to the time passed to AVAssetWriter.startSessionAtSourceTime to determine the timing of the frame

in the output file.

Returns true if the pixel buffer was successfully appended, otherwise false.

If the operation was unsuccessful, you might invoke the AVAssetWriter object's finishWriting method in order to save a partially completed asset.
Do not modify a CVPixelBuffer or its contents after you have passed it to this method.

4.20.5 assetWriterInput as AVAssetWriterInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset writer input to which the adaptor should append pixel buffers. (read-only)

4.20.6 assetWriterInputPixelBufferAdaptorWithAssetWriterInput(input as AVAssetWriterInputMBS, sourcePixelBufferAttributes as dictionary) as AVAssetWriterInputPixelBufferAdaptorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new pixel buffer adaptor to receive pixel buffers for writing to the output file.
Notes: input: The asset writer input to which the receiver should append pixel buffers.
Currently, only asset writer inputs that accept media data of type AVMediaTypeVideo can be used to initialize a pixel buffer adaptor.
It is an error to pass a sample buffer input that is already attached to another instance of AVAssetWriterInputPixelBufferAdaptor.

sourcePixelBufferAttributes: The attributes of pixel buffers that will be vended by the input's CVPixelBufferPool.
Pixel buffer attributes keys for the pixel buffer pool are defined in <CoreVideo/CVPixelBuffer.h>. To take advantage of the improved efficiency of appending buffers created from the adaptor's pixel buffer pool, you should specify pixel buffer attributes that most closely accommodate the source format of the video frames being appended.
Pass nil if you do not need a pixel buffer pool for allocating buffers.

Returns a new pixel buffer adaptor to receive pixel buffers for writing to the output file.

To specify the pixel format type, the pixelBufferAttributes dictionary should contain a value for kCVPixelBufferPixelFormatTypeKey. For example, use kCVPixelFormatType_32BGRA as Integer for 8-bit-per-channel BGRA, or use kCVPixelFormatType_420YpCbCr8BiPlanarVideoRange as Integer for 2-plane YCbCr.

4.20.7 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.20.8 Constructor(input as AVAssetWriterInputMBS, sourcePixelBufferAttributes as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a new pixel buffer adaptor to receive pixel buffers for writing to the output file.
Notes: input: The asset writer input to which the receiver should append pixel buffers.
Currently, only asset writer inputs that accept media data of type AVMediaTypeVideo can be used to initialize a pixel buffer adaptor.
It is an error to pass a sample buffer input that is already attached to another instance of AVAssetWriterInputPixelBufferAdaptor.

sourcePixelBufferAttributes: The attributes of pixel buffers that will be vended by the input's CVPixelBufferPool.
Pixel buffer attributes keys for the pixel buffer pool are defined in <CoreVideo/CVPixelBuffer.h>. To take advantage of the improved efficiency of appending buffers created from the adaptor's pixel buffer pool, you should specify pixel buffer attributes that most closely accommodate the source format of the video frames being appended.
Pass nil if you do not need a pixel buffer pool for allocating buffers.

Returns a new pixel buffer adaptor to receive pixel buffers for writing to the output file.

To specify the pixel format type, the pixelBufferAttributes dictionary should contain a value for kCVPixelBufferPixelFormatTypeKey. For example, use kCVPixelFormatType_32BGRA as Integer for 8-bit-per-channel BGRA, or use kCVPixelFormatType_420YpCbCr8BiPlanarVideoRange as Integer for 2-plane YCbCr.

4.20.9 sourcePixelBufferAttributes as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The pixel buffer attributes of pixel buffers that will be vended by the adaptor's CVPixelBufferPool. (read-only)
Notes: The value of this property is a dictionary containing pixel buffer attributes keys defined in <CoreVideo/CVPixelBuffer.h>.

4.20.10 Properties

4.20.11 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.21 class AVAssetWriterMBS

4.21.1 class AVAssetWriterMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVAssetWriter object to write media data to a new file of a specified audiovisual container type, such as a QuickTime movie file or an MPEG-4 file, with support for automatic interleaving of media data for multiple concurrent tracks.
Notes: You can get the media data for one or more assets from instances of AVAssetReader or even from outside the AV Foundation API set. Media data is presented to AVAssetWriter for writing in the form of CMSampleBuffers. Sequences of sample data appended to the asset writer inputs are considered to fall within "sample-writing sessions." You must call startSessionAtSourceTime to begin one of these sessions.

Using AVAssetWriter, you can optionally re-encode media samples as they are written. You can also optionally write metadata collections to the output file.

You can only use a given instance of AVAssetWriter once to write to a single file. If you want to write to files multiple times, you must use a new instance of AVAssetWriter each time.
Xojo Developer Magazine

- 12.2, page 47: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.21.2 Methods

4.21.3 addInput(input as AVAssetWriterInputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds an input to the receiver.
Notes: input: The asset writer input to be added.

Inputs are created with a media type and output settings. These both must be compatible with the receiver. You cannot add inputs after writing has started.

4.21.4 addInputGroup(inputGroup as AVAssetWriterInputGroupMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Adds an asset writer input group instance to the asset writer.
Notes: inputGroup: The asset writer input group to be added.

The asset writer will mark the tracks associated with grouped inputs as mutually exclusive to each other for playback or other processing, if the output container format supports mutually exclusive relationships among tracks.
When an input group is added to an asset writer, the value of marksOutputTrackAsEnabled for the AVAssetWriterInput instance set as the default input will automatically be set to true and all other inputs in the group will be set to false.
Input groups cannot be added after writing has started.
Available in OS X v10.9 and later.

4.21.5 assetWriterWithFile(outputFile as folderitem, outputFileType as string, byref error as NSErrorMBS) as AVAssetWriterMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset writer for writing to the file identified by a given folderitem in a format specified by a given UTI.
Notes: outputFile: The location of the file to be written.
outputFileType: The UTI-identified format of the file to be written.
For example, AVFileTypeQuickTimeMovie for a QuickTime movie file, AVFileTypeMPEG4 for an MPEG-4 file, and AVFileTypeAMR for an adaptive multi-rate audio format file.
Error: If initialization of the asset writer fails, upon return contains an error object that describes the problem.

Returns an asset writer for writing to the file identified by URL in the format specified by outputFileType, or nil if the writer could not be initialized.
Writing will fail if a file already exists at URL. UTIs for container formats that can be written are declared in AVMediaFormat.h.

4.21.6 assetWriterWithURL(outputURL as string, outputFileType as string, byref error as NSErrorMBS) as AVAssetWriterMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset writer for writing to the file identified by a given URL in a format specified by a given UTI.
Notes: outputURL: The location of the file to be written. The URL must be a file URL.
outputFileType: The UTI-identified format of the file to be written.
For example, AVFileTypeQuickTimeMovie for a QuickTime movie file, AVFileTypeMPEG4 for an MPEG-4 file, and AVFileTypeAMR for an adaptive multi-rate audio format file.
Error: If initialization of the asset writer fails, upon return contains an error object that describes the problem.

Returns an asset writer for writing to the file identified by URL in the format specified by outputFileType,

or nil if the writer could not be initialized.
Writing will fail if a file already exists at URL. UTIs for container formats that can be written are declared in AVMediaFormat.h.

4.21.7 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.21.8 availableMediaTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The media types for which inputs can be added (read-only)
Notes: Some media types may not be accepted within the type of file with which the writer was initialized.

4.21.9 canAddInput(input as AVAssetWriterInputMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given input can be added to the receiver.
Notes: input: The asset writer input to be tested.

Returns true if input can be added, otherwise false.
You cannot add an input that accepts media data of a type that is not compatible with the receiver, or with output settings that are not compatible with the receiver.

4.21.10 canAddInputGroup(input as AVAssetWriterInputGroupMBS) as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns whether an input group can be added to the receiver.
Notes: inputGroup: The asset writer input group to be added.
Returns true if inputGroup can be added to the inputGroups, otherwise false.

If outputFileType specifies a container format that does not support mutually exclusive relationships among tracks, or if the specified instance of AVAssetWriterInputGroup contains inputs with media types that cannot be related, the group cannot be added.

Available in OS X v10.9 and later.

4.21.11 canApplyOutputSettings(outputSettings as dictionary, mediaType as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether give output settings are supported for a specified media type.
Notes: outputSettings: The output settings to validate.
mediaType: The media type for which the output settings are validated.

Returns true if the output settings in outputSettings are supported for mediaType, otherwise false.
You can use this method to test, for example, whether video output settings that specify H.264 compression will fail (as would be the case if the container format for which the writer was initialized does not support the carriage of H.264-compressed video).

4.21.12 cancelWriting

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the writer to cancel writing.
Notes: This method blocks until writing is canceled.

4.21.13 Constructor(outputFile as folderitem, outputFileType as string, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset writer for writing to the file identified by a given URL in a format specified by a given UTI.
Notes: outputFile: The location of the file to be written.
outputFileType: The UTI-identified format of the file to be written.
For example, AVFileTypeQuickTimeMovie for a QuickTime movie file, AVFileTypeMPEG4 for an MPEG-4 file, and AVFileTypeAMR for an adaptive multi-rate audio format file.
Error: If initialization of the asset writer fails, upon return contains an error object that describes the problem.

Returns an asset writer for writing to the file identified by URL in the format specified by outputFileType, or nil if the writer could not be initialized.

Writing will fail if a file already exists at URL. UTIs for container formats that can be written are declared in AVMediaFormat.h.
See also:

- 4.21.14 Constructor(outputURL as string, outputFileType as string, byref error as NSErrorMBS) 231

4.21.14 Constructor(outputURL as string, outputFileType as string, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset writer for writing to the file identified by a given URL in a format specified by a given UTI.
Notes: outputURL: The location of the file to be written. The URL must be a file URL.
outputFileType: The UTI-identified format of the file to be written.
For example, AVFileTypeQuickTimeMovie for a QuickTime movie file, AVFileTypeMPEG4 for an MPEG-4 file, and AVFileTypeAMR for an adaptive multi-rate audio format file.
Error: If initialization of the asset writer fails, upon return contains an error object that describes the problem.

Returns an asset writer for writing to the file identified by URL in the format specified by outputFileType, or nil if the writer could not be initialized.

Writing will fail if a file already exists at URL. UTIs for container formats that can be written are declared in AVMediaFormat.h.
See also:

- 4.21.13 Constructor(outputFile as folderitem, outputFileType as string, byref error as NSErrorMBS) 230

4.21.15 endSessionAtSourceTime(endTime as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Concludes an explicit sample-writing session.
Notes: endTime: The ending asset time for the sample-writing session, in the timeline of the source samples.

You may invoke this method to complete a session you began by invoking startSessionAtSourceTime.

You do not need to call this method; if you call finishWriting without calling this method, the session's effective end time will be the latest end timestamp of the session's samples (that is, no samples will be edited out at the end).

The endTime defines the moment on the timeline of source samples at which the session ends. In the case of the QuickTime movie file format, each sample-writing session's startTime,Ä¶endTime pair corresponds to a period of movie time into which the session's samples are inserted. Samples with later timestamps will be still be added to the media but will be edited out of the movie. So if the first session has duration D1 = endTime - startTime, it will be inserted into the movie at movie time 0 through D1; the second session would be inserted into the movie at movie time D1 through D1+D2, and so on.

It is legal to have a session with no samples; this will cause creation of an empty edit of the prescribed duration.

4.21.16 error as NSErrorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: If the receiver's status is AVAssetWriterStatusFailed, describes the error that caused the failure. (read-only)
Notes: The value of this property is an error object that describes what caused the receiver to no longer be able to write to its output file. If the receiver's status is not AVAssetWriterStatusFailed, the value of this property is nil.

4.21.17 finishWriting as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Completes the writing of the output file.
Notes: Returns true if writing can be finished, otherwise false.

This method blocks until writing is finished. When this method returns successfully, the file being written by the receiver is complete and ready to use. You can check the values of the status and error properties for more information on why writing could not be finished.

4.21.18 finishWritingWithCompletionHandler(tag as Variant = nil)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Marks all unfinished inputs as finished and completes the writing of the output file.
Notes: Calls later AVFoundationMBS.finishWritingCompleted event.

This method returns immediately and causes its work to be performed asynchronously. To determine whether the operation succeeded, you can check the value of the status property within the handler parameter. If the status is AVAssetWriterStatusFailed, then the error property will contain an instance of NSError that describes the failure.

To guarantee that all sample buffers are successfully written, you must ensure that all calls to appendSampleBuffer and appendPixelBuffer have returned before invoking this method.

Available in OS X v10.9 and later.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.21.19 inputGroups as AVAssetWriterInputGroupMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array of asset writer input groups that have been added to the asset writer. (read-only)
Notes: The value of this property is an array containing concrete instances of AVAssetWriterInputGroup.
Input groups are added to the receiver using the addInputGroup method.
Available in OS X v10.9 and later.

4.21.20 inputs as AVAssetWriterInputMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset writer inputs associated with the asset writer. (read-only)
Notes: The array contains AVAssetWriterInput objects.

4.21.21 metadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The collection of metadata for association with the asset and for carriage in the output file.
Notes: The array contains AVMetadataItem objects.
See also SetMetaData.

4.21.22 movieFragmentInterval as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time to elapse between writing movie fragments.
Notes: This property only applies to the QuickTime movie file type.

Sometimes a write operation may be unexpectedly interrupted (because a process crashes, for example). By using movie fragments, such a partially-written QuickTime movie file can be successfully opened and played up to the largest multiple of movieFragmentInterval smaller than the point at which the write operation was interrupted.

The default value is kCMTimeInvalid, which means that movie fragments should not be used, that only a movie atom describing all of the media in the file should be written.

You cannot set the value after writing has started.

4.21.23 movieTimeScale as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies the asset-level time scale to be used.
Notes: For file types that contain a moov atom, such as QuickTime Movie files, specifies the asset-level time scale to be used.
The default value is 0, which indicates that you should choose a convenient value, if applicable.
You cannot set the value after writing has started.

4.21.24 outputFileType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The file format of the writer's output. (read-only)
Notes: The format is identified by the UTI, specified when the writer is initialized.

4.21.25 outputURL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL to which output is directed. (read-only)
Notes: The URL is the same as that specified when the writer is initialized.

4.21.26 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Set the collection of metadata for association with the asset and for carriage in the output file.
Notes: The array contains AVMetadataItem objects.
You cannot set the value after writing has started.

4.21.27 shouldOptimizeForNetworkUse as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the output file should be written in way that makes it more suitable for playback over a network.
Notes: When the value of this property is true, the output file will be written in such a way that playback can start after only a small amount of the file is downloaded.

The default value is false.
You cannot set the value after writing has started.

4.21.28 startSessionAtSourceTime(startTime as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initiates a sample-writing session for the output asset.
Notes: startTime: The starting asset time for the sample-writing session, in the timeline of the source samples.

Sequences of sample data appended to the asset writer inputs are considered to fall within "sample-writing sessions." You must call this method to begin one of these sessions.

Each writing session has a start time which, where allowed by the file format being written, defines the mapping from the timeline of source samples onto the file's timeline. In the case of the QuickTime movie file format, the first session begins at movie time 0, so a sample appended with timestamp T will be played at movie time (T-startTime). Samples with timestamps before startTime will still be added to the output media but will be edited out of the movie. If the earliest buffer for an input is later than startTime, an empty edit will be inserted to preserve synchronization between tracks of the output asset.

It is an error to invoke this method twice in a row without invoking endSessionAtSourceTime in between.

4.21.29 startWriting as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the writer to start writing its output.
Notes: Returns true if writing can be started, otherwise false.

You must call this method after all inputs have added and other configuration properties have been set to tell the receiver to prepare for writing. After invoking this method, you can start writing sessions using startSessionAtSourceTime and can write media samples using the methods provided by each of the writer's inputs.

status signals the terminal state of the asset reader, and if a failure occurs, error describes the failure.

4.21.30 status as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The status of writing samples to the receiver's output file. (read-only)
Notes: The value of this property is an AVAssetWriterStatus constant that indicates whether writing is in progress, has completed successfully, has been canceled, or has failed. If an attempt to append samples fails, you can check the value of this property to determine why no more samples could be written.

4.21.31 Properties

4.21.32 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.21.33 Constants

Status Constants

Constant	Value	Description
AVAssetWriterStatusCancelled	4	Cancelled
AVAssetWriterStatusCompleted	2	Completed
AVAssetWriterStatusFailed	3	Failed
AVAssetWriterStatusUnknown	0	Unknown
AVAssetWriterStatusWriting	1	Writing

4.22 class AVAsynchronousVideoCompositionRequestMBS

4.22.1 class AVAsynchronousVideoCompositionRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An AVAsynchronousVideoCompositionRequest instance contains the information necessary for a video compositor to render an output pixel buffer.
Notes: The video compositor must implement the AVVideoCompositing protocol.
Available in OS X v10.9 and later.

4.22.2 Methods

4.22.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9.

4.22.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The constructor.

4.22.5 copy as AVAsynchronousVideoCompositionRequestMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.22.6 finishCancelledRequest

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Invoked by the custom compositor when the composition request was canceled.
Notes: Available in OS X v10.9 and later.

4.22.7 finishWithComposedVideoFrame(composedVideoFrame as CVPixelBufferMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Invoked by the custom compositor when the composition request succeeded.
Notes: composedVideoFrame: The successfully composed pixel buffer.
Available in OS X v10.9 and later.

4.22.8 finishWithError(error as NSErrorMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Invoked by the custom compositor when the composition request failed.
Notes: error: Returns the error encountered during the compositing.
Available in OS X v10.9 and later.

4.22.9 sourceFrameByTrackID(trackID as Integer) as CVPixelBufferMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the source pixel buffer for the specified track ID.
Notes: trackID: The track ID for the requested source frame.
Returns the source CVPixelBufferRef for trackID.
Available in OS X v10.9 and later.

4.22.10 sourceTrackIDs as Integer()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The track IDs of all the source buffers that are available to compose the frame. (read-only)
Notes: Available in OS X v10.9 and later.

4.22.11 Properties

4.22.12 compositionTime as CMTimeMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The time for which the frame should be composed. (read-only)
Notes: Available in OS X v10.9 and later.

(Read only property)

4.22.13 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.22.14 renderContext as AVVideoCompositionRenderContextMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The video composition render context making the request. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.22.15 videoCompositionInstruction as AVVideoCompositionInstructionMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The video composition instruction to use to compose the frame. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.23 class AVAudioMixInputParametersMBS

4.23.1 class AVAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVAudioMixInputParameters object represents the parameters that should be applied to an audio track when it is added to a mix.
Notes: Audio volume is currently supported as a time-varying parameter. AVAudioMixInputParameters has a mutable subclass, AVMutableAudioMixInputParameters.

You use an instance AVAudioMixInputParameters to apply audio volume ramps for an input to an audio mix. Mix parameters are associated with audio tracks via the trackID property.

Before the first time at which a volume is set, a volume of 1.0 used; after the last time for which a volume has been set, the last volume is used. Within the time range of a volume ramp, the volume is interpolated between the start volume and end volume of the ramp. For example, setting the volume to 1.0 at time 0 and also setting a volume ramp from a volume of 0.5 to 0.2 with a timeRange of [4.0, 5.0] results in an audio volume parameters that hold the volume constant at 1.0 from 0.0 sec to 4.0 sec, then cause it to jump to 0.5 and descend to 0.2 from 4.0 sec to 9.0 sec, holding constant at 0.2 thereafter.

4.23.2 Methods

4.23.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.23.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.

4.23.5 copy as AVAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.23.6 getVolumeRampForTime(time as CMTimeMBS, byref startVolume as Double, byref endVolume as Double, byref timeRange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Obtains the volume ramp that includes the specified time.
Notes: time: If a ramp with a time range that contains the specified time has been set, information about the effective ramp for that time is supplied. Otherwise, information about the first ramp that starts after the specified time is supplied.
startVolume: A variable to receive the starting volume value for the volume ramp.
endVolume: A variable to receive the ending volume value for the volume ramp.
timeRange: A variable to a CMTimeRange to receive the time range of the volume ramp.

Returns true if the values were retrieved successfully, otherwise false. Returns false if time is beyond the duration of the last volume ramp that has been set.

4.23.7 mutableCopy as AVMutableAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an editable copy of the object.

4.23.8 trackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The ID of the audio track to which the parameters should be applied. (read-only)

4.23.9 Properties

4.23.10 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.24 class AVAudioMixMBS

4.24.1 class AVAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVAudioMix object manages the input parameters for mixing audio tracks.
Notes: It allows custom audio processing to be performed on audio tracks during playback or other operations.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr1

4.24.2 Methods

4.24.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.24.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.24.5 copy as AVAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.24.6 inputParameters as AVAudioMixInputParametersMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The parameters for inputs to the mix (read-only)
Notes: The array contains instances of AVAudioMixInputParameters. Note that an instance of AVAudioMixInputParameters is not required for each audio track that contributes to the mix; audio for those

without associated AVAudioMixInputParameters objects will be included in the mix, processed according to default behavior.

4.24.7 mutableCopy as AVMutableAudioMixMBS

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Creates a mutable copy.

4.24.8 Properties

4.24.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.25 class AVAudioPlayerMBS

4.25.1 class AVAudioPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An instance of the AVAudioPlayer class, called an audio player, provides playback of audio data from a file or memory.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp3")
dim e as NSErrorMBS
dim u as new AVAudioPlayerMBS(f,e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox str(u.duration)+" seconds"
end if
```

Notes: Apple recommends that you use this class for audio playback unless you are playing audio captured from a network stream or require very low I/O latency. For an overview of audio technologies, see Audio & Video Starting Point and "Using Audio" in Multimedia Programming Guide.

Using an audio player you can:

- Play sounds of any duration
- Play sounds from files or memory buffers
- Loop sounds
- Play multiple sounds simultaneously, one sound per audio player, with precise synchronization
- Control relative playback level, stereo positioning, and playback rate for each sound you are playing
- Seek to a particular point in a sound file, which supports such application features as fast forward and rewind
- Obtain data you can use for playback-level metering

The AVAudioPlayer class lets you play sound in any audio format available in OS X. The plugin provides the events for this audio player in AVFoundationMBS class.

To play, pause, or stop an audio player, call one of its playback control methods, described in "Configuring and Controlling Playback."

To configure an appropriate audio session for playback, refer to AVAudioSession Class Reference and AVAudioSessionDelegate Protocol Reference. To learn how your choice of file formats impacts the simultaneous playback of multiple sounds, refer to "iPhone Hardware and Software Audio Codecs" in Multimedia Programming Guide.
Blog Entries

- MBS Xojo Plugins, version 20.0pr5
- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr6
- MBS Xojo Plugins, version 18.2pr8
- MBS Xojo / Real Studio Plugins, version 14.5pr4
- MBS Xojo / Real Studio Plugins, version 14.2pr4

Xojo Developer Magazine

- 16.5, page 9: News
- 12.2, page 46: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.25.2 Methods

4.25.3 audioPlayerWithData(Data as MemoryBlock, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated memory buffer.
Notes: data: A block of data containing a sound to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see "Using Audio" in Multimedia Programming Guide.
Error: Pass in the address of a nil-initialized NSError object. If an error occurs, upon return the NSError object describes the error. If you do not want error information, pass in NULL.

On success, you get an initialized AVAudioPlayer object. Else the handle property is 0 and the Error parameter contains a code that describes the problem.
See also:

- 4.25.4 audioPlayerWithData(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 246
- 4.25.5 audioPlayerWithData(Data as String, byref error as NSErrorMBS) as AVAudioPlayerMBS 246
- 4.25.6 audioPlayerWithData(Data as String, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 247

4.25.4 audioPlayerWithData(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified data and file type hint.
Notes: data: The data object containing the audio.
fileTypeHintUtiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs.
error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that data that may possibly be corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without filetypehint on older versins.
See also:

- 4.25.3 audioPlayerWithData(Data as MemoryBlock, byref error as NSErrorMBS) as AVAudioPlayerMBS 245
- 4.25.5 audioPlayerWithData(Data as String, byref error as NSErrorMBS) as AVAudioPlayerMBS 246
- 4.25.6 audioPlayerWithData(Data as String, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 247

4.25.5 audioPlayerWithData(Data as String, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated memory buffer.
Notes: data: A block of data containing a sound to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see "Using Audio" in Multimedia Programming Guide.
Error: Pass in the address of a nil-initialized NSError object. If an error occurs, upon return the NSError object describes the error. If you do not want error information, pass in NULL.

On success, you get an initialized AVAudioPlayer object. Else the handle property is 0 and the Error parameter contains a code that describes the problem.
See also:

- 4.25.3 audioPlayerWithData(Data as MemoryBlock, byref error as NSErrorMBS) as AVAudioPlayerMBS 245
- 4.25.4 audioPlayerWithData(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 246

- 4.25.6 audioPlayerWithData(Data as String, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 247

4.25.6 audioPlayerWithData(Data as String, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified data and file type hint.
Notes: data: The data object containing the audio.
fileTypeHintUtiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs.
error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that data that may possibly be corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without filetypehint on older versins.
See also:

- 4.25.3 audioPlayerWithData(Data as MemoryBlock, byref error as NSErrorMBS) as AVAudioPlayerMBS 245
- 4.25.4 audioPlayerWithData(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 246
- 4.25.5 audioPlayerWithData(Data as String, byref error as NSErrorMBS) as AVAudioPlayerMBS 246

4.25.7 audioPlayerWithFile(File as folderitem, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated sound file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp3")
dim e as NSErrorMBS
dim u as AVAudioPlayerMBS.audioPlayerWithFile(f,e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox str(u.duration)+" seconds"
end if
```

Notes: File: A folderitem identifying the sound file to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see ”Using Audio” in Multimedia Programming Guide.
Error: If an error occurs, upon return the NSError object describes the error..

On success, you have an initialized AVAudioPlayer object. On failure, the Error parameter contains a code that describes the problem.
See also:

- 4.25.8 audioPlayerWithFile(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 248

4.25.8 audioPlayerWithFile(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified URL and file type hint.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio.
utiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs. See AVFileType* strings.
Error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that files which are not self identifying, or possibly even corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without hint for older versions.
See also:

- 4.25.7 audioPlayerWithFile(File as folderitem, byref error as NSErrorMBS) as AVAudioPlayerMBS 247

4.25.9 audioPlayerWithURL(URL as string, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated sound file.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see ”Using Audio” in Multimedia Programming Guide.
Error: If an error occurs, upon return the NSError object describes the error..

On success, you have an initialized AVAudioPlayer object. On failure, the Error parameter contains a code that describes the problem.

See also:

- 4.25.10 audioPlayerWithURL(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 249

4.25.10 audioPlayerWithURL(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) as AVAudioPlayerMBS

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified URL and file type hint.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio.
utiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs. See AVFileType* strings.
Error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that files which are not self identifying, or possibly even corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without hint for older versions.
See also:

- 4.25.9 audioPlayerWithURL(URL as string, byref error as NSErrorMBS) as AVAudioPlayerMBS 248

4.25.11 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.25.12 averagePowerForChannel(channel as Integer) as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the average power for a given channel, in decibels, for the sound being played.
Notes: channelNumber: The audio channel whose average power value you want to obtain. Channel numbers are zero-indexed. A monaural signal, or the left channel of a stereo signal, has channel number 0.

Returns a floating-point representation, in decibels, of a given audio channel's current average power. A return value of 0 dB indicates full scale, or maximum power; a return value of -160 dB indicates minimum power (that is, near silence).

If the signal provided to the audio player exceeds ¬±full scale, then the return value may exceed 0 (that is, it may enter the positive range).

To obtain a current average power value, you must call the updateMeters method before calling this method.

4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated memory buffer.
Notes: data: A block of data containing a sound to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see "Using Audio" in Multimedia Programming Guide.
Error: Pass in the address of a nil-initialized NSError object. If an error occurs, upon return the NSError object describes the error. If you do not want error information, pass in NULL.

On success, you get an initialized AVAudioPlayer object. Else the handle property is 0 and the Error parameter contains a code that describes the problem.
See also:

- 4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) 250
- 4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS) 251
- 4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) 252
- 4.25.17 Constructor(URL as string, byref error as NSErrorMBS) 252
- 4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) 253

4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified data and file type hint.
Notes: data: The data object containing the audio.
fileTypeHintUtiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs.
error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that data that may possibly be corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without filetypehint on older versins.
See also:

- 4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS) 250
- 4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS) 251
- 4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) 252
- 4.25.17 Constructor(URL as string, byref error as NSErrorMBS) 252
- 4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) 253

4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated sound file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp3")
dim e as NSErrorMBS
dim u as new AVAudioPlayerMBS(f,e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox str(u.duration)+" seconds"
end if
```

Notes: File: A folderitem identifying the sound file to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see "Using Audio" in Multimedia Programming Guide.
Error: If an error occurs, upon return the NSError object describes the error..

On success, you have an initialized AVAudioPlayer object. On failure, the Error parameter contains a code that describes the problem.
See also:

- 4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS) 250
- 4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) 250

- 4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) 252
- 4.25.17 Constructor(URL as string, byref error as NSErrorMBS) 252
- 4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) 253

4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified URL and file type hint.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio.
utiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs. See AVFileType* strings.
Error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that files which are not self identifying, or possibly even corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without hint for older versions.
See also:

- 4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS) 250
- 4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) 250
- 4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS) 251
- 4.25.17 Constructor(URL as string, byref error as NSErrorMBS) 252
- 4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) 253

4.25.17 Constructor(URL as string, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player for playing a designated sound file.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio. For a list of supported formats, see ”Using Audio” in Multimedia Programming Guide.
Error: If an error occurs, upon return the NSError object describes the error..

On success, you have an initialized AVAudioPlayer object. On failure, the Error parameter contains a code that describes the problem.
See also:

- 4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS) 250
- 4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) 250
- 4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS) 251
- 4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) 252
- 4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS) 253

4.25.18 Constructor(URL as string, fileTypeHintUtiString as string, byref error as NSErrorMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Initializes and returns an audio player using the specified URL and file type hint.
Notes: url: A URL identifying the sound file to play. The audio data must be in a format supported by Core Audio.
utiString: A UTI that is used as a file type hint. The supported UTIs are defined in File Format UTIs. See AVFileType* strings.
Error: If an error occurs, upon return the NSError object describes the error.

The utiString file type hint tells the parser what kind of sound data to expect so that files which are not self identifying, or possibly even corrupt, can be successfully parsed.

Available in OS X v10.9 and later, but plugin calls variant without hint for older versions.
See also:

- 4.25.13 Constructor(Data as MemoryBlock, byref error as NSErrorMBS) 250
- 4.25.14 Constructor(Data as MemoryBlock, fileTypeHintUtiString as string, byref error as NSErrorMBS) 250
- 4.25.15 Constructor(File as folderitem, byref error as NSErrorMBS) 251
- 4.25.16 Constructor(File as folderitem, fileTypeHintUtiString as string, byref error as NSErrorMBS) 252
- 4.25.17 Constructor(URL as string, byref error as NSErrorMBS) 252

4.25.19 pause

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Pauses playback; sound remains ready to resume playback from where it left off.
Notes: Calling pause leaves the audio player prepared to play; it does not release the audio hardware that

was acquired upon calling play or prepareToPlay.

4.25.20 peakPowerForChannel(channel as Integer) as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the peak power for a given channel, in decibels, for the sound being played.
Notes: channelNumber: The audio channel whose peak power value you want to obtain. Channel numbers are zero-indexed. A monaural signal, or the left channel of a stereo signal, has channel number 0.

Returns a floating-point representation, in decibels, of a given audio channel's current peak power. A return value of 0 dB indicates full scale, or maximum power; a return value of -160 dB indicates minimum power (that is, near silence).

If the signal provided to the audio player exceeds ¬±full scale, then the return value may exceed 0 (that is, it may enter the positive range).

To obtain a current peak power value, you must call the updateMeters method before calling this method.

4.25.21 play as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Plays a sound asynchronously.
Notes: Returns true on success, or false on failure.
Calling this method implicitly calls the prepareToPlay method if the audio player is not already prepared to play.

4.25.22 playAtTime(time as Double = 0.0) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Plays a sound asynchronously, starting at a specified point in the audio output device's timeline.
Example:

```
dim p as AVAudioPlayerMBS // your player

if p.playAtTime(p.deviceCurrentTime + 5.0) then
// playing in 5 seconds
else
// failed
end if
```

Notes: time: The number of seconds to delay playback, relative to the audio output device's current time.

Returns true on success, or false on failure.

To learn about the virtual audio output device's timeline, read the description for the deviceCurrentTime property.
Calling this method implicitly calls the prepareToPlay method if the audio player is not already prepared to play.

4.25.23 prepareToPlay as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prepares the audio player for playback by preloading its buffers.
Notes: Returns true on success, or false on failure.

Calling this method preloads buffers and acquires the audio hardware needed for playback, which minimizes the lag between calling the play method and the start of sound output.
Calling the stop method, or allowing a sound to finish playing, undoes this setup.

4.25.24 stop

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Stops playback and undoes the setup needed for playback.
Notes: Calling this method, or allowing a sound to finish playing, undoes the setup performed upon calling the play or prepareToPlay methods.

The stop method does not reset the value of the currentTime property to 0. In other words, if you call stop during playback and then call play, playback resumes at the point where it left off.

4.25.25 updateMeters

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Refreshes the average and peak power values for all channels of an audio player.
Notes: To obtain current audio power values, you must call this method before calling averagePowerForChannel or peakPowerForChannel.

4.25.26 Properties

4.25.27 currentDevice as String

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The UID of the current audio device (as a string).
Notes: Available on MacOS 10.13 or newer.
Settable in v20.0 or newer.
(Read and Write property)

4.25.28 currentTime as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The playback point, in seconds, within the timeline of the sound associated with the audio player.
Notes: If the sound is playing, currentTime is the offset of the current playback position, measured in seconds from the start of the sound. If the sound is not playing, currentTime is the offset of where playing starts upon calling the play method, measured in seconds from the start of the sound.

By setting this property you can seek to a specific point in a sound file or implement audio fast-forward and rewind functions.
(Read and Write property)

4.25.29 data as MemoryBlock

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The data object containing the sound associated with the audio player. (read-only)
Notes: Returns nil if the audio player has no data (that is, if it was not initialized with an Memoryblock).
(Read only property)

4.25.30 deviceCurrentTime as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time value, in seconds, of the audio output device. (read-only)
Notes: The value of this property increases monotonically while an audio player is playing or paused.

If more than one audio player is connected to the audio output device, device time continues incrementing as long as at least one of the players is playing or paused.

If the audio output device has no connected audio players that are either playing or paused, device time reverts to 0.

Use this property to indicate "now" when calling the playAtTime instance method. By configuring multiple audio players to play at a specified offset from deviceCurrentTime, you can perform precise synchronization—as described in the discussion for that method.
(Read only property)

4.25.31 duration as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the total duration, in seconds, of the sound associated with the audio player. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp3")
dim e as NSErrorMBS
dim u as new AVAudioPlayerMBS(f,e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox str(u.duration)+" seconds"
end if
```

Notes: (Read only property)

4.25.32 enableRate as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that specifies whether playback rate adjustment is enabled for an audio player.
Notes: To enable adjustable playback rate for an audio player, set this property to YES after you initialize the player and before you call the prepareToPlay instance method for the player.
Available in OS X v10.8 and later.
(Read and Write property)

4.25.33 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.25.34 isPlaying as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the audio player is playing (true) or not (false). (read-only)
Notes: To find out when playback has stopped, use the AVFoundationMBS.audioPlayerDidFinishPlaying event.

Important: Do not poll this property (that is, do not use it inside of a loop) in an attempt to discover when playback has stopped.
(Read only property)

4.25.35 meteringEnabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that specifies the audio-level metering on/off state for the audio player.
Notes: The default value for the meteringEnabled property is off (Boolean false). Before using metering for an audio player, you need to enable it by setting this property to true.
(Read and Write property)

4.25.36 Name as String

Plugin Version: 18.3, Platform: macOS, Targets: All.

Function: The name of the audio file.
Notes: This uses the URL to pull the display name if possible, so you can show it to user or identify the player when debugging.
(Read only property)

4.25.37 numberOfChannels as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The number of audio channels in the sound associated with the audio player. (read-only)
Notes: (Read only property)

4.25.38 numberOfLoops as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The number of times a sound will return to the beginning, upon reaching the end, to repeat playback.
Notes: A value of 0, which is the default, means to play the sound once. Set a positive integer value to specify the number of times to return to the start and play again. For example, specifying a value of 1 results in a total of two plays of the sound. Set any negative integer value to loop the sound indefinitely until you call the stop method.
(Read and Write property)

4.25.39 pan as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The audio player's stereo pan position.
Notes: By setting this property you can position a sound in the stereo field. A value of –1.0 is full left, 0.0 is center, and 1.0 is full right.
(Read and Write property)

4.25.40 rate as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The audio player's playback rate.
Example:

```
dim s as sound // your sound
dim a as AVAudioPlayerMBS = s.AVAudioPlayerMBS
a.rate = 0.5
```

Notes: This property's default value of 1.0 provides normal playback rate. The available range is from 0.5 for half-speed playback through 2.0 for double-speed playback.

To set an audio player's playback rate, you must first enable rate adjustment as described in the enableRate property description.
Available in OS X v10.8 and later.

(Read and Write property)

4.25.41 settings as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The audio player's settings dictionary, containing information about the sound associated with the player. (read-only)
Notes: An audio player's settings dictionary contains keys for the following information about the player's associated sound:

Channel layout (AVChannelLayoutKey)
Encoder bit rate (AVEncoderBitRateKey)
Audio data format (AVFormatIDKey)
Channel count (AVNumberOfChannelsKey)
Sample rate (AVSampleRateKey)

The settings keys are described in AV Foundation Audio Settings Constants.
(Read only property)

4.25.42 URL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL for the sound associated with the audio player. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp3")
dim e as NSErrorMBS
dim u as new AVAudioPlayerMBS(f,e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox u.URL
end if
```

Notes: Returns "" if the audio player was not initialized with a URL.
(Read only property)

4.25.43 volume as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The playback gain for the audio player, ranging from 0.0 through 1.0.
Notes: (Read and Write property)

4.26 class AVAudioRecorderMBS

4.26.1 class AVAudioRecorderMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An instance of the AVAudioRecorder class, called an audio recorder, provides audio recording capability in your application.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.caf”)
dim recordSetting as new Dictionary

dim n as Integer = OSTypeFromStringMBS(AVFoundationMBS.kAudioFormatLinearPCM)
recordSetting.Value(AVFoundationMBS.AVFormatIDKey) = n
recordSetting.Value(AVFoundationMBS.AVSampleRateKey) = 44100.0
recordSetting.Value(AVFoundationMBS.AVNumberOfChannelsKey) = 2
recordSetting.Value(AVFoundationMBS.AVLinearPCMBitDepthKey) = 16
recordSetting.Value(AVFoundationMBS.AVLinearPCMIsBigEndianKey) = false
recordSetting.Value(AVFoundationMBS.AVLinearPCMIsFloatKey) = false


dim error as NSErrorMBS
dim recorder as new AVAudioRecorderMBS(f, recordSetting, error)

if error<>Nil then
MsgBox error.LocalizedDescription
Return
end if

if recorder.recordForDuration(10) then
// ok
else
MsgBox ”Failed to start recording.”
end if
```

Notes: Using an audio recorder you can:

- Record until the user stops the recording
- Record for a specified duration
- Pause and resume a recording
- Obtain input audio-level data that you can use to provide level metering

In OS X, the audio comes from the system's default audio input device as set by a user in System Preferences.

You can implement a subclass of AVFoundationMBS to respond to audio interruptions and audio decoding errors, and to the completion of a recording.

To configure a recording, including options such as bit depth, bit rate, and sample rate conversion quality, configure the audio recorder's settings dictionary. Use the settings keys described in AV Foundation Audio Settings Constants.

To configure an appropriate audio session for recording, refer to AVAudioSession Class Reference and AVAudioSessionDelegate Protocol Reference.

The AVAudioRecorder class is intended to allow you to make audio recordings with very little programming overhead. Other classes that can be used for recording audio in OS X include AVCaptureAudioDataOutput and the Audio Queue services described in the Audio Queue Services Programming Guide.

In OS X, you can also use the AVCaptureAudioFileOutput class to record audio.
Available in OS X v10.7 and later.

Calls audioRecorderEncodeErrorDidOccur and audioRecorderDidFinishRecording events on AVFoundationMBS class.

4.26.2 Methods

4.26.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.26.4 averagePowerForChannel(channelNumber as Integer) as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the average power for a given channel, in decibels, for the sound being recorded.
Notes: channelNumber: The number of the channel that you want the average power value for.
Returns the current average power, in decibels, for the sound being recorded. A return value of 0 dB indicates full scale, or maximum power; a return value of -160 dB indicates minimum power (that is, near silence).

If the signal provided to the audio recorder exceeds ¬±full scale, then the return value may exceed 0 (that is, it may enter the positive range).

To obtain a current average power value, you must call the updateMeters method before calling this method.

4.26.5 Constructor(file as folderitem, settings as Dictionary, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an audio recorder.
Notes: url: The file system location to record to. The file type to record to is inferred from the file extension included in this parameter's value.
settings: Settings for the recording session. For information on the settings available for an audio recorder, see AV Foundation Audio Settings Constants.
Error: Pass in the address of a nil-initialized NSError object. If an error occurs, upon return the NSError object describes the error. If you do not want error information, pass in NULL.
See also:

- 4.26.6 Constructor(URL as string, settings as Dictionary, byref error as NSErrorMBS) 264

4.26.6 Constructor(URL as string, settings as Dictionary, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an audio recorder.
Notes: url: The file system location to record to. The file type to record to is inferred from the file extension included in this parameter's value.
settings: Settings for the recording session. For information on the settings available for an audio recorder, see AV Foundation Audio Settings Constants.
Error: Pass in the address of a nil-initialized NSError object. If an error occurs, upon return the NSError object describes the error. If you do not want error information, pass in NULL.
See also:

- 4.26.5 Constructor(file as folderitem, settings as Dictionary, byref error as NSErrorMBS) 264

4.26.7 currentTime as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time, in seconds, since the beginning of the recording. (read-only)
Notes: When the audio recorder is stopped, calling this method returns a value of 0.

Available in OS X v10.7 and later.

4.26.8 deleteRecording as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Deletes a recorded audio file.
Notes: Returns true on success, or false on failure.
The audio recorder must be stopped before you call this method.

4.26.9 pause

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Pauses a recording.
Notes: Call record to resume recording.

4.26.10 peakPowerForChannel(channelNumber as Integer) as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the peak power for a given channel, in decibels, for the sound being recorded.
Notes: channelNumber: The number of the channel that you want the peak power value for.

Returns the current peak power, in decibels, for the sound being recorded. A return value of 0 dB indicates full scale, or maximum power; a return value of -160 dB indicates minimum power (that is, near silence).

If the signal provided to the audio recorder exceeds ¬±full scale, then the return value may exceed 0 (that is, it may enter the positive range).

To obtain a current peak power value, call the updateMeters method immediately before calling this method.

4.26.11 prepareToRecord as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an audio file and prepares the system for recording.
Notes: Returns true on success, or false on failure.

Creates an audio file at the location specified by the url parameter in the Constructor method. If a file already exists at that location, this method overwrites it.
The preparation invoked by this method takes place automatically when you call record. Use prepareToRecord when you want recording to start as quickly as possible upon calling record.

4.26.12 record as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Starts or resumes recording.
Notes: Returns true on success, or false on failure.
Calling this method implicitly calls prepareToRecord, which creates (or erases) an audio file and prepares the system for recording.

4.26.13 recordForDuration(duration as Double) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Records for a specified duration of time.
Notes: duration: The maximum duration, in seconds, for the recording.
Returns true on success, or false on failure.

The recorder stops when the duration of recorded audio reaches the value in the duration parameter.

Calling this method implicitly calls prepareToRecord, which creates (or erases) an audio file and prepares the system for recording.

4.26.14 Recording as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the audio recorder is recording (true), or not (false). (read-only)

4.26.15 settings as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The audio settings for the audio recorder. (read-only)
Notes: Audio recorder settings are in effect only after you explicitly call the prepareToRecord method, or

after you call it implicitly by starting recording. The audio settings keys are described in AV Foundation Audio Settings Constants.

4.26.16 stop

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Stops recording and closes the audio file.

4.26.17 updateMeters

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Refreshes the average and peak power values for all channels of an audio recorder.
Notes: To obtain current audio power values, you must call this method before you call averagePowerForChannel or peakPowerForChannel.

4.26.18 url as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL for the audio file associated with the audio recorder. (read-only)

4.26.19 Properties

4.26.20 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.26.21 MeteringEnabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether audio-level metering is enabled (true), or not (false).
Notes: By default, audio level metering is off for an audio recorder. Because metering uses computing resources, turn it on only if you intend to use it.

Available in OS X v10.7 and later.
(Read and Write computed property)

4.27 class AVCaptureAudioChannelMBS

4.27.1 class AVCaptureAudioChannelMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVCaptureAudioChannel to monitor the average and peak power levels in an audio channel in a capture connection (see AVCaptureConnection).
Notes: An AVCaptureConnection object from an input producing audio to an output receiving audio exposes an array of AVCaptureAudioChannel objects, one for each channel of audio available. You can poll for audio levels by iterating through these audio channel objects.
You cannot create instances of AVCaptureAudioChannel directly.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.27.2 Methods

4.27.3 averagePowerLevel as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The instantaneous average power level, in dB. (read-only)

4.27.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.27.5 peakHoldLevel as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The peak hold power level, in dB. (read-only)

4.27.6 Properties

4.27.7 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.27.8 Enabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the channel is currently enabled for data capture.
Notes: By default, all AVCaptureAudioChannel objects exposed by a connection are enabled. You may set enabled to false to stop the flow of data for a particular channel.
(Read and Write computed property)

4.27.9 volume as Double

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The current volume (gain) of the channel.
Notes: The volume property indicates the current volume or gain of the receiver as a floating point value between 0.0 and 1.0. If you want to boost the gain in software, you may specify a a value greater than 1.0.
(Read and Write computed property)

4.28 class AVCaptureAudioDataOutputMBS

4.28.1 class AVCaptureAudioDataOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureAudioDataOutput is a concrete sub-class of AVCaptureOutput that you use, via its delegate, to process audio sample buffers from the audio being captured.
Notes: Subclass of the AVCaptureOutputMBS class.

4.28.2 Methods

4.28.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
Notes: Enables events on the new object.

4.28.4 EnableEvents

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables the event for an existing audio data output object.
Notes: For the audio data output, the plugin will with enabled events call captureOutputDidOutputSampleBuffer event.

4.28.5 Properties

4.28.6 audioSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The settings used to decode or re-encode audio before it is output.
Notes: The value of this property is a dictionary containing values for audio settings keys defined in AVAudioSettings.h.
If the value of this property is nil, samples are output in their device native format.
(Read and Write computed property)

4.29 class AVCaptureAudioFileOutputMBS

4.29.1 class AVCaptureAudioFileOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: AVCaptureMovieFileOutput is a concrete sub-class of AVCaptureFileOutput that writes captured audio to any audio file type supported by CoreAudio.
Notes: AVCaptureAudioFileOutput implements the complete file recording interface declared by AVCaptureFileOutput for writing media data to audio files. In addition, you can configure options specific to the audio file formats, including writing metadata collections to each file and specifying audio encoding options. AVCaptureAudioFileOutput does not, however, support startRecordingToOutputFileURL use startRecordingToOutputFileURL with outputfiletype parameter instead.
Subclass of the AVCaptureFileOutputMBS class.

4.29.2 Methods

4.29.3 availableOutputFileTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns array containing UTIs identifying the file types AVCaptureAudioFileOutput can write.

4.29.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

4.29.5 EnableEvents

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables the events for this output.
Notes: Those are normally handled automatically, but if you add event with AddHandler you need to call this method.

4.29.6 metadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A collection of metadata to be written to the receiver's output files.
Notes: The value of this property is an array of AVMetadataItem objects representing the collection of top-level metadata to be written in each output file. Only ID3 v2.2, v2.3, or v2.4 style metadata items are supported.

4.29.7 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the metadata array.

4.29.8 startRecordingToOutputFile(file as folderitem, outputFileType as string)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tells the receiver to start recording to a new file of the specified format, and specifies a delegate that will be notified when recording is finished.
Notes: file: The output file. If a file already exists when capturing starts, recording to the new file will fail. fileType: A UTI indicating the format of the file to be written. UTIs for common audio file types are declared in AVMediaFormat.h.

You do not need not to call stopRecording before calling this method while another recording is in progress. If this method is invoked while an existing output file was already being recorded, no media samples will be discarded between the old file and the new file.

When recording is stopped—by calling stopRecording, by changing files using this method, or because of an error—the remaining data that needs to be included to the file will be written in the background. Therefore, you must specify a delegate that will be notified when all data has been written to the file using the captureOutputDidFinishRecordingToOutputFileAtURL method. The recording delegate can also optionally implement methods that inform it when data starts being written, when recording is paused and resumed, and when recording is about to be finished.

On OS X, if this method is called within the captureOutputDidOutputSampleBuffer event method, the first samples written to the new file are guaranteed to be those contained in the sample buffer passed to that method.

4.29.9 startRecordingToOutputFileURL(URL as string, outputFileType as string)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tells the receiver to start recording to a new file of the specified format, and specifies a delegate

that will be notified when recording is finished.
Notes: URL: The URL of the output file. This method throws an NSInvalidArgumentException if the URL is not a valid file URL. If a file at the given URL already exists when capturing starts, recording to the new file will fail.
fileType: A UTI indicating the format of the file to be written. UTIs for common audio file types are declared in AVMediaFormat.h.

You do not need not to call stopRecording before calling this method while another recording is in progress. If this method is invoked while an existing output file was already being recorded, no media samples will be discarded between the old file and the new file.

When recording is stopped—by calling stopRecording, by changing files using this method, or because of an error—the remaining data that needs to be included to the file will be written in the background. Therefore, you must specify a delegate that will be notified when all data has been written to the file using the captureOutputDidFinishRecordingToOutputFileAtURL method. The recording delegate can also optionally implement methods that inform it when data starts being written, when recording is paused and resumed, and when recording is about to be finished.

On OS X, if this method is called within the captureOutputDidOutputSampleBuffer event method, the first samples written to the new file are guaranteed to be those contained in the sample buffer passed to that method.

4.29.10 Properties

4.29.11 audioSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The settings used to decode or re-encode audio before it is output by the receiver.
Notes: The value of this property is a dictionary containing values for audio settings keys defined in AVAudioSettings.h. If you set the value of this property to nil, the output vends samples in their device native format.
(Read and Write computed property)

4.30 class AVCaptureAudioPreviewOutputMBS

4.30.1 class AVCaptureAudioPreviewOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: AVCaptureAudioPreviewOutput is a concrete subclass of AVCaptureOutput that you use to preview audio being captured.
Notes: Instances of AVCaptureAudioPreviewOutput are associated with a Core Audio output device that can be used to play audio being captured by the capture session. You can obtain the unique ID of a Core Audio device using its kAudioDevicePropertyDeviceUID property.
Subclass of the AVCaptureOutputMBS class.

4.30.2 Methods

4.30.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The default constructor.

4.30.4 Properties

4.30.5 outputDeviceUniqueID as string

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the unique ID of the Core Audio output device being used to play preview audio.
Notes: The value of this property is an string containing the unique ID of the Core Audio device to be used for output, or "" if the default system output should be used.
(Read and Write computed property)

4.30.6 volume as Double

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the preview volume of the output.
Notes: The value of this property is the preview volume of the receiver, where 1.0 is the maximum volume and 0.0 is muted.
(Read and Write computed property)

4.31 class AVCaptureConnectionMBS

4.31.1 class AVCaptureConnectionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVCaptureConnection object represents a connection between capture input and capture output objects associated with a capture session.
Notes: Capture inputs (instances of AVCaptureInput) have one or more input ports (instances of AVCaptureInputPort). Capture outputs (instances of AVCaptureOutput) can accept data from one or more sources (for example, an AVCaptureMovieFileOutput object accepts both video and audio data).

You can only add an AVCaptureConnection instance to a session using addConnection: if canAddConnection: returns true. When using addInput or addOutput, connections are formed automatically between all compatible inputs and outputs. You only need to add connections manually when adding an input or output with no connections. You can also use connections to enable or disable the flow of data from a given input or to a given output.

4.31.2 Methods

4.31.3 audioChannels as AVCaptureAudioChannelMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVCaptureAudioChannel objects. (read-only)
Notes: This property is only applicable to connections involving audio.

4.31.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.31.5 connectionWithInputPort(port as AVCaptureInputPortMBS, layer as AVCaptureVideoPreviewLayerMBS) as AVCaptureConnectionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a capture connection describing a connection between the specified input port and the specified video preview layer.
Notes: port: An AVCaptureInputPort object associated with an AVCaptureInput object.

layer: An AVCaptureVideoPreviewLayer object.

Returns a capture connection describing a connection between the port and layer.

You can add the connection returned by this method to an AVCaptureSession instance using addConnection:.

When using addInput or addOutput, connections are automatically formed between all compatible inputs and outputs. You do not need to manually create and add connections to the session unless you use the primitive addInputWithNoConnections or addOutputWithNoConnections methods.

4.31.6 connectionWithInputPorts(ports() as AVCaptureInputPortMBS, output as AVCaptureOutputMBS) as AVCaptureConnectionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a capture connection describing a connection between the specified input ports and the specified output.
Notes: ports: An array of AVCaptureInputPort objects associated with AVCaptureInput objects.
output: An AVCaptureOutput object.

Returns a capture connection describing a connection between inputPorts and the output.

You can add the connection returned by this method to an AVCaptureSession instance using addConnection.

When using addInput or addOutput, connections are automatically formed between all compatible inputs and outputs. You do not need to manually create and add connections to the session unless you use the primitive addInputWithNoConnections or addOutputWithNoConnections methods.

4.31.7 Constructor(port as AVCaptureInputPortMBS, layer as AVCaptureVideoPreviewLayerMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a capture connection to describe a connection between the specified input port and the specified video preview layer.
Notes: port: An AVCaptureInputPort object associated with an AVCaptureInput object.
layer: An AVCaptureVideoPreviewLayer object.

Returns a capture connection describing a connection between the port and layer.

You can add the connection returned by this method to an AVCaptureSession instance using addConnection.

When using addInput or addOutput, connections are automatically formed between all compatible inputs and outputs. You do not need to manually create and add connections to the session unless you use the primitive addInputWithNoConnections or addOutputWithNoConnections methods.
See also:

- 4.31.8 Constructor(ports() as AVCaptureInputPortMBS, output as AVCaptureOutputMBS) 278

4.31.8 Constructor(ports() as AVCaptureInputPortMBS, output as AVCaptureOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a capture connection to describe a connection between the specified input ports and the specified output.
Notes: ports: An array of AVCaptureInputPort objects associated with AVCaptureInput objects.
output: An AVCaptureOutput object.

Returns acapture connection initialized to describe a connection between inputPorts and the output.

You can add the connection returned by this method to an AVCaptureSession instance using addConnection.

When using addInput or addOutput, connections are automatically formed between all compatible inputs and outputs. You do not need to manually create and add connections to the session unless you use the primitive addInputWithNoConnections or addOutputWithNoConnections methods.
See also:

- 4.31.7 Constructor(port as AVCaptureInputPortMBS, layer as AVCaptureVideoPreviewLayerMBS) 277

4.31.9 inputPorts as AVCaptureInputPortMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The connection's input ports. (read-only)
Notes: Input ports are instances of AVCaptureInputPort.

4.31.10 Properties

4.31.11 Active as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the connection is active. (read-only)
Notes: (Read only property)

4.31.12 automaticallyAdjustsVideoMirroring as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the value of videoMirrored can change based on configuration of the session.
Notes: For some session configurations, video data flowing through the connection will be mirrored by default. When the value of this property is true, the value of videoMirrored may change depending on the configuration of the session, for example after switching to a different capture device input.

The default value is true.
(Read and Write property)

4.31.13 Enabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the connection is enabled.
Notes: (Read and Write property)

4.31.14 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.31.15 output as AVCaptureOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The connection's output port. (read-only)
Notes: (Read only property)

4.31.16 videoFieldMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An indicator of how interlaced video flowing through the connection should be treated.
Notes: This property is only applicable to connections involving video.
You can only set this value if supportsVideoFieldMode is true.
(Read and Write property)

4.31.17 VideoFieldModeSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A Boolean value that indicates whether the connection supports setting the videoFieldMode property. (read-only)
Notes: This property is only applicable to connections involving video.
(Read only property)

4.31.18 videoMaxFrameDuration as CMTimeMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The maximum time interval between which the receiver should output consecutive video frames.
Notes: The value of this property specifies the maximum duration of each video frame output by the connection, placing an upper bound on the amount of time that should separate consecutive frames. The value is equivalent to the reciprocal of the minimum frame rate.
A value of kCMTimeZero or kCMTimeInvalid indicates an unlimited minimum frame rate.
The default value is kCMTimeInvalid.
You can only set this value if supportsVideoMaxFrameDuration is true.
Available in OS X v10.9 and later.
(Read and Write property)

4.31.19 VideoMaxFrameDurationSupported as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the connection supports setting the videoMaxFrameDuration property. (read-only)

Notes: This property is only applicable to connections involving video.
Available in OS X v10.9 and later.
(Read only property)

4.31.20 videoMinFrameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The minimum time interval between which the receiver should output consecutive video frames.
Notes: The value of this property specifies the minimum duration of each video frame output by the connection, placing a lower bound on the amount of time that should separate consecutive frames. The value is equivalent to the reciprocal of the maximum frame rate.

A value of kCMTimeZero or kCMTimeInvalid indicates an unlimited maximum frame rate.

The default value is kCMTimeInvalid.

You can only set this value if supportsVideoMinFrameDuration is true.
(Read and Write property)

4.31.21 VideoMinFrameDurationSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the connection supports setting the videoMinFrameDuration property. (read-only)
Notes: This property is only applicable to connections involving video.
(Read only property)

4.31.22 VideoMirrored as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the video flowing through the connection should be mirrored about its vertical axis.
Notes: This property is only applicable to connections involving video.

if the value of supportsVideoMirroring is true, you can set videoMirrored to true to flip the video about its vertical axis and produce a mirror-image effect.
(Read and Write property)

4.31.23 VideoMirroringSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the connection supports video mirroring. (read-only)
Notes: (Read only property)

4.31.24 videoOrientation as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The orientation of the video.
Notes: This property is only applicable to connections involving video.

If the value of supportsVideoOrientation is true, you can set videoOrientation to rotate the video buffers being consumed by the connection's output. Setting videoOrientation does not necessarily result in a physical rotation of video buffers. For example, a video connection to an AVCaptureMovieFileOutput object handles orientation using a Quicktime track matrix; using an AVCaptureStillImageOutput object, orientation is handled using Exif tags.
(Read and Write property)

4.31.25 VideoOrientationSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the connection supports changing the orientation of the video. (read-only)
Notes: (Read only property)

4.31.26 videoPreviewLayer as AVCaptureVideoPreviewLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video preview layer associated with the connection. (read-only)
Notes: This property is the set if you initialized the connection using Constructor or connectionWithInputPort.
(Read only property)

4.31.27 Constants

Video Orientation Constants

Constant	Value	Description
AVCaptureVideoOrientationLandscapeLeft	4	Indicates that video should be oriented horizontally, top on th
AVCaptureVideoOrientationLandscapeRight	3	Indicates that video should be oriented horizontally, top on th
AVCaptureVideoOrientationPortrait	1	Indicates that video should be oriented vertically, top at the t
AVCaptureVideoOrientationPortraitUpsideDown	2	Indicates that video should be oriented vertically, top at the b

Video Field Modes

Constant	Value	Description
AVVideoFieldModeBoth	0	Indicates that both top and bottom video fields in interlaced content should be passed through.
AVVideoFieldModeBottomOnly	2	Indicates that the bottom video field only in interlaced content should be passed through.
AVVideoFieldModeDeinterlace	3	Indicates that top and bottom video fields in interlaced content should be deinterlaced.
AVVideoFieldModeTopOnly	1	Indicates that only the top video field in interlaced content should be passed through.

4.32 class AVCaptureDeviceFormatMBS

4.32.1 class AVCaptureDeviceFormatMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The class for device format information.
Notes: An AVCaptureDeviceFormat object wraps a CMFormatDescription (see CMFormatDescriptionRef) and other format-related information, such as minimum and maximum frame rate.

An AVCaptureDeviceFormat object is immutable. AVCaptureDevice uses AVCaptureDeviceFormat objects to describe supported formats and the active format of an instance.
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 22.2
- MBS Xojo / Real Studio Plugins, version 13.2pr4

4.32.2 Methods

4.32.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.32.4 DisplayName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The display text for this object.
Notes: The plugin formats a string with the device format for displaying.

4.32.5 formatDescription as CMFormatDescriptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A CMFormatDescription describing an AVCaptureDevice active or supported format. (read-only).

4.32.6 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An string describing the media type of an AVCaptureDevice active or supported format. (read-only)
Notes: Supported media types are listed in AVMediaFormat.h.

4.32.7 videoSupportedFrameRateRanges as AVFrameRateRangeMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the format's supported frame rate ranges/ (read-only)
Notes: The value is an array of AVFrameRateRange objects, one for each of the format's supported video frame rate ranges.

4.32.8 Properties

4.32.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.32.10 videoMaxZoomFactor as Double

Plugin Version: 22.2, Platform: macOS, Targets: iOS only.

Function: Indicates the maximum zoom factor allowed for the format.
Notes: A maximum factor of 1.0 indicates that the format isn,Äôt capable of zooming.
(Read only property)

4.33 class AVCaptureDeviceInputMBS

4.33.1 class AVCaptureDeviceInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureDeviceInput is a concrete sub-class of AVCaptureInput you use to capture data from an AVCaptureDevice object.
Notes: Subclass of the AVCaptureInputMBS class.

4.33.2 Methods

4.33.3 Constructor(Device as AVCaptureDeviceMBS, byref error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an input to use a specified device.
Notes: device: The device from which to capture input.
Error: If an error occurs during initialization, upon return contains an NSErrorMBS object describing the problem.

4.33.4 device as AVCaptureDeviceMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The device with which the input is associated. (read-only)

4.33.5 deviceInputWithDevice(device as AVCaptureDeviceMBS, byref error as NSErrorMBS) as AVCaptureDeviceInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an input initialized to use a specified device.
Notes: device: The device from which to capture input.
Error: If an error occurs during initialization, upon return contains an NSError object describing the problem.

Returns an input initialized to use device.

4.34 class AVCaptureDeviceInputSourceMBS

4.34.1 class AVCaptureDeviceInputSourceMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An AVCaptureDeviceInputSource object represents a distinct input source on an AVCaptureDevice object.
Notes: An AVCaptureDevice object may optionally present an array of input sources, representing distinct mutually exclusive inputs to the device. For example, an audio capture device might have ADAT optical and analog input sources; a video capture device might have an HDMI input source, or a component input source.
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 21.0pr6

4.34.2 Methods

4.34.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

4.34.4 inputSourceID as string

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The input source ID. (read-only)
Notes: The ID is unique among the input sources exposed by a given AVCaptureDevice object.

4.34.5 localizedName as string

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A localized, human-readable, name for the input source. (read-only)
Notes: You can use this property to display the name of the capture device input source in a user interface.

4.34.6 Properties

4.34.7 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

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

4.35 class AVCaptureDeviceMBS

4.35.1 class AVCaptureDeviceMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVCaptureDevice object represents a physical capture device and the properties associated with that device.
Example:

```
dim device as AVCaptureDeviceMBS = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox device.localizedName
```

Notes: You use a capture device to configure the properties of the underlying hardware. A capture device also provides input data (such as audio or video) to an AVCaptureSession object.

You use the methods of the AVCaptureDevice class to enumerate the available devices, query their capabilities, and be informed about when devices come and go. Before you attempt to set properties of a capture device (its focus mode, exposure mode, and so on), you must first acquire a lock on the device using the lockForConfiguration method. You can then set the properties and release the lock using the unlockForConfiguration method. You may hold the lock if you want all settable device properties to remain unchanged. However, holding the device lock unnecessarily may degrade capture quality in other applications sharing the device and is not recommended.

May work for all built-in cameras and microphones as well as external USB webcams, as far as Apple supports them.
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 23.5
- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5
- MBS Xojo Plugins, version 23.5pr6
- MBS Xojo Plugins, version 22.2pr7
- MBS Xojo Plugins, version 20.2pr4
- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.4
- MBS Xojo Plugins, version 18.4pr7

Xojo Developer Magazine

- 22.1, page 9: News

- 17.1, page 10: News
- 16.6, page 9: News

4.35.2 Methods

4.35.3 authorizationStatusForMediaType(mediaType as string) as Integer

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Returns the client's authorization status for accessing the underlying hardware that supports a given media type.
Notes: Mediatype: The media type, either AVMediaTypeVideo or AVMediaTypeAudio.
Returns the authorization status of the client

This method returns the AVAuthorizationStatus of the client for accessing the underlying hardware supporting the media type. Media type constants are defined in AVFoundationMBS. If any media type other than AVMediaTypeVideo or AVMediaTypeAudio is supplied, an NSInvalidArgumentException will be thrown. If the status is AVAuthorizationStatusNotDetermined, you may use the requestAccessForMediaType method to request access by prompting the user.

see
https://developer.apple.com/documentation/avfoundation/cameras_and_media_capture/requesting_authorization_for_media_capture_on_macos?language=objc

4.35.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.35.5 availableReactionTypes as String()

Plugin Version: 23.5, Platform: macOS, Targets: All.

Function: Returns a list of reaction types which can be passed to performEffectForReaction.
Example:

```
dim device as AVCaptureDeviceMBS // your device

MessageBox Join(device.availableReactionTypes, EndOfLine)
```

Notes: The list may differ between devices, or be affected by changes to active format.

Requires macOS 14.0 or iOS 17.0.

List seen on macOS 14.0:

ReactionBalloons
ReactionFireworks
ReactionThumbsUp
ReactionRain
ReactionThumbsDown
ReactionLasers
ReactionConfetti
ReactionHeart

You may check AVCaptureReactionEffectStateMBS.available for whether this is available.

4.35.6 AVCaptureMaxAvailableTorchLevel as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The maximum torch level.
Notes: This constant always represents the maximum available torch level, independent of the actual maximum value currently supported by the device. Thus, pass this constant to the setTorchModeOnWithLevel:error: in situations where you want to specify the maximum torch level without having to worry about whether the device is overheating and might not accept a value of 1.0 as the maximum.
Available in OS X v10.9 and later.

4.35.7 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.35.8 defaultDeviceWithMediaType(mediaType as string) as AVCaptureDeviceMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the default device used to capture data of a given media type.

Example:

```
dim device as AVCaptureDeviceMBS = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeAudio)
MsgBox device.localizedName
```

Notes: mediaType: A media type identifier.
For possible values, see AV Foundation Constants Reference.

4.35.9 devices as AVCaptureDeviceMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the available capture devices on the system.
Example:

```
dim devices() as AVCaptureDeviceMBS = AVCaptureDeviceMBS.devices
dim names() as string

for each d as AVCaptureDeviceMBS in devices
names.Append d.localizedName
next

// show all names in a message dialog
MsgBox Join(names, EndOfLine)
```

4.35.10 devicesWithMediaType(mediaType as string) as AVCaptureDeviceMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the devices able to capture data of a given media type.
Example:

```
dim devices() as AVCaptureDeviceMBS = AVCaptureDeviceMBS.devicesWithMediaType(AVFoundationMBS.AVMediaTypeAudio)
dim names() as string

for each d as AVCaptureDeviceMBS in devices
names.Append d.localizedName
next

// show all names of audio devices in a message dialog
MsgBox Join(names, EndOfLine)
```

Notes: mediaType: A media type identifier.
For possible values, see AV Foundation Constants Reference.

4.35.11 deviceWithUniqueID(deviceUniqueID as string) as AVCaptureDeviceMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the device with a given ID.
Notes: deviceUniqueID: The ID of a capture device.

4.35.12 formats as AVCaptureDeviceFormatMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVCaptureDeviceFormat objects representing the formats supported by the device (read-only)
Notes: You can use this property to enumerate the formats natively supported by the receiver.
You can set modelID to any of the formats in this array.
You can observe changes to the value of this property using key-value observing.

4.35.13 hasMediaType(mediaType as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the device provides media with a given type.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.hasMediaType(AVFoundationMBS.AVMediaTypeVideo)) // true
MsgBox str(device.hasMediaType(AVFoundationMBS.AVMediaTypeAudio)) // false
```

Notes: mediaType: A media type, such as AVMediaTypeVideo, AVMediaTypeAudio, or AVMediaTypeMuxed. For a complete list of supported media type constants, see AV Foundation Constants Reference.

Returns true if the device provides media of type mediaType, otherwise false.

4.35.14 inputSources as AVCaptureDeviceInputSourceMBS()

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An array of AVCaptureDeviceInputSource objects representing the input sources supported by the device. (read-only)
Notes: Some devices can capture data from one of multiple data sources (different input jacks on the same audio device, for example). For devices with multiple possible data sources, you can use this property to enumerate the possible choices.

You can observe changes to the value of this property using key-value observing.

4.35.15 isExposureModeSupported(exposureMode as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the given exposure mode is supported.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isExposureModeSupported(device.AVCaptureExposureModeAutoExpose))
```

Notes: exposureMode: An exposure mode. See "AVCaptureExposureMode" for possible values.
Returns true if exposureMode is supported, otherwise false.

4.35.16 isFlashModeSupported(FlashMode as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the given flash mode is supported.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isFlashModeSupported(device.AVCaptureFlashModeAuto))
```

Notes: flashMode: A flash mode. See "AVCaptureFlashMode" for possible values.
Returns true if flashMode is supported, otherwise false.

4.35.17 isFocusModeSupported(focusMode as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the given focus mode is supported.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isFocusModeSupported(device.AVCaptureFocusModeContinuousAutoFocus))
```

Notes: focusMode: A focus mode. See "AVCaptureFocusMode" for possible values.

Returns true if focusMode is supported, otherwise false.

4.35.18 isTorchModeSupported(torchMode as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the device supports the specified torch mode.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isTorchModeSupported(device.AVCaptureTorchModeAuto))
```

Notes: torchMode: The desired torch mode. For a list of possible values, see "AVCaptureTorchMode."
Returns true if torchMode is supported, otherwise false.

4.35.19 isWhiteBalanceModeSupported(mode as Integer) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the given white balance mode is supported.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isWhiteBalanceModeSupported(device.AVCaptureWhiteBalanceModeAutoWhiteBalance))
```

Notes: whiteBalanceMode: A focus mode. See "AVCaptureWhiteBalanceMode" for possible values.

Returns true if whiteBalanceMode is supported, otherwise false.

4.35.20 linkedDevices as AVCaptureDeviceMBS()

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: An array of AVCaptureDevice objects representing the devices physically linked to the receiver. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeAudio)

dim linkedDevices() as AVCaptureDeviceMBS
linkedDevices = device.linkedDevices

for each d as AVCaptureDeviceMBS in linkedDevices
MsgBox device.localizedName+" ,Üí "+d.localizedName
next
```

Notes: The value of this property is an array of AVCaptureDevice objects that are a part of the same physical device as the receiver. For example, for an external iSight camera, the array contains an AVCaptureDevice instance representing the external iSight microphone.

4.35.21 lockForConfiguration(byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Requests exclusive access to the device's hardware properties.
Notes: Error: If a lock cannot be acquired, this pointer contains an NSError object that describes the problem.

Returns true if a lock was acquired or false if it was not.

You must call this method before attempting to configure the hardware related properties of the device. This method returns true when it successfully locks the device for configuration by your code. After configuring the device properties, call unlockForConfiguration to release the configuration lock and allow other apps to make changes.

You may hold onto a lock (instead of releasing it) if you require the device properties to remain unchanged. However, holding the device lock unnecessarily may degrade capture quality in other apps sharing the device.

4.35.22 performEffectForReaction(reactionType as String)

Plugin Version: 23.5, Platform: macOS, Targets: All.

Function: Triggers a specified reaction on the video stream.
Example:

```
dim device as AVCaptureDeviceMBS // your device

device.performEffectForReaction AVCaptureReactionEffectStateMBS.AVCaptureReactionTypeBalloons
```

Notes: reactionType: Indicates which reaction to perform.

The entries in reactionEffectsInProgress may not reflect one-to-one against calls to this method. Depending on reaction style or resource limits, triggering multiple overlapping reactions of the same type may be coalesced into extending an existing reaction rather than overlaying a new one.

The reactionType requested must be one of those listed in availableReactionTypes or an exception will be thrown. Performing a reaction when canPerformReactionEffects is false is ignored, and VoIP applications are encouraged to transmit and display such reactions outside of the video feed.

Requires macOS 14.0 or iOS 17.0.
You may check AVCaptureReactionEffectStateMBS.available for whether this is available.

4.35.23 reactionEffectsInProgress as AVCaptureReactionEffectStateMBS()

Plugin Version: 23.5, Platform: macOS, Targets: All.

Function: Contains an array of reaction effects that are currently being performed by the device, sorted by timestamp.
Example:

```
dim device as AVCaptureDeviceMBS // your device

dim effects() as AVCaptureReactionEffectStateMBS = device.reactionEffectsInProgress

for each effect as AVCaptureReactionEffectStateMBS in effects
MessageBox effect.reactionType
next
```

Notes: If observing old and new values in the KVO callback, the reaction effects which are still running in the new array will have kCMTimeInvalid as their endTime property. Reaction effects which have ended will only be in the old array, and will have their endTime property set to the presentation time of the first frame where the reaction effect was no longer present.

Reaction effects which are triggered by either a call to performEffectForReaction or by the automatic gesture detection will be reflected in this array. It is key-value observable to be notified when reaction effects begin or end.

Requires macOS 14.0 or iOS 17.0.
You may check AVCaptureReactionEffectStateMBS.available for whether this is available.

4.35.24 requestAccessForMediaType(mediaType as string, tag as variant = nil)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Requests access to the underlying hardware for the media type, showing a dialog to the user if necessary.
Notes: mediaType: The media type, either AVMediaTypeVideo or AVMediaTypeAudio

Use this function to request access to the hardware for a given media type. Media type constants are defined in AVFoundationMBS
If any media type other than AVMediaTypeVideo or AVMediaTypeAudio is supplied, an NSInvalidArgumentException will be thrown.

This call will not block while the user is being asked for access, allowing the client to continue running. Until access has been granted, any AVCaptureDevices for the media type will vend silent audio samples or black video frames. The user is only asked for permission the first time the client requests access. Later calls use the permission granted by the user.

Note that the authorization dialog will automatically be shown if the status is AVAuthorizationStatusNotDetermined when creating an AVCaptureDeviceInput.

Invoking this method with AVMediaTypeAudio is equivalent to calling AVAudioSessionMBS.requestRecordPermission.

Call returns directly as request is performed on thread.
The requestAccessForMediaTypeCompleted event is called later in your AVFoundationMBS subclass with results on main thread.
The granted value will always true for MacOS 10.13 or older.

see
https://developer.apple.com/documentation/avfoundation/cameras_and_media_capture/requesting_authorization_for_media_capture_on_macos?language=objc

4.35.25 setTransportControlsPlaybackMode(mode as Integer, speed as Double)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the transport control's playback mode and speed.
Notes: mode: An AVCaptureDeviceTransportControlsPlaybackMode constant indicating whether the deck should be put into play mode.
speed: A float value indicating the speed at which to wind or play the tape.

Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, this method raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

4.35.26 supportsAVCaptureSessionPreset(preset as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver can be used in an capture session configured with the given preset.
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.supportsAVCaptureSessionPreset(AVFoundationMBS.AVCaptureSessionPreset1280x720))
```

Notes: preset: A capture session preset.

Returns true if the receiver can be used with preset, otherwise false.
An AVCaptureSession instance can be associated with a preset that configures its inputs and outputs to fulfill common use cases. You can use this method to determine if the receiver can be used in a capture session with the given preset. For a list of preset constants, see AVCaptureSession Class Reference.

4.35.27 unlockForConfiguration

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Relinquishes exclusive control over the device's configuration.
Notes: Call this method to release the lock acquired using the lockForConfiguration method when you are done configuring the device.

4.35.28 Properties

4.35.29 activeFormat as AVCaptureDeviceFormatMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The currently active format of the receiver.
Notes: You use this property to get or set the currently active device format.

To set the format, you must first obtain exclusive access to the receiver using focusMode. If you do not obtain exclusive access, setActiveFormat throws an NSGenericException.

The set method, setActiveFormat, throws an NSInvalidArgumentException if you pass a format not present in the supportsAVCaptureSessionPreset array.

You can observe changes to the value of this property using key-value observing.
(Read and Write property)

4.35.30 activeInputSource as AVCaptureDeviceInputSourceMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The currently active input source.
Notes: You use this property to get or set the currently active input source.

To set the format, you must first obtain exclusive access to the receiver using focusMode. If you do not obtain exclusive access, setActiveInputSource throws an NSGenericException.

The set method, setActiveInputSource, throws an NSInvalidArgumentException if you pass a format not present in the activeInputSource array.

You can observe changes to the value of this property using key-value observing.
(Read and Write property)

4.35.31 activeVideoMaxFrameDuration as CMTimeMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The currently active maximum frame duration.
Notes: A device's maximum frame duration is the reciprocal of its minimum frame rate. You can set the value of this property to limit the minimum frame rate during a capture session. The capture device auto-

matically chooses a default maximum frame duration based on its active format. After changing the value of this property, you can return to the default maximum frame duration by setting this property's value to kCMTimeInvalid. Choosing a new preset for the capture session also resets this property to its default value.

Attempting to set this property to a value not found in the active format's videoSupportedFrameRateRanges array raises an exception (NSInvalidArgumentException).

Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. Otherwise, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using Key-value observing.
Available in OS X v10.9 and later.
(Read and Write property)

4.35.32 activeVideoMinFrameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The currently active minimum frame duration.
Notes: You use this property to get or set the currently active minimum frame duration.

The value of this property is the reciprocal of its maximum frame rate. To limit the maximum frame rate, you can set the value of this property to a value supported by the receiver's modelID (see videoSupportedFrameRateRanges).

Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

The device object throws NSInvalidArgumentException if you try to assign an unsupported value to this property.

You can observe changes to the value of this property using key-value observing.
(Read and Write property)

4.35.33 canPerformReactionEffects as Boolean

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Indicates whether reactions can be performed on a particular AVCaptureDevice.
Example:

```
dim device as AVCaptureDeviceMBS // your device
if device <>nil then
MessageBox "canPerformReactionEffects: "+device.canPerformReactionEffects.ToString
end if
```

Notes: This requires reactionEffectsEnabled to be true, as well as using a AVCaptureDeviceFormat with reactionEffectsSupported.

This readonly property returns YES when resources for reactions are available on the device instance. When true, calls to performEffectForReaction: will render on the video feed, otherwise those calls are ignored. It is key-value observable.

Requires macOS 14.0 or iOS 17.0.
You may check AVCaptureReactionEffectStateMBS.available for whether this is available.
(Read only property)

4.35.34 exposureMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The exposure mode for the device.
Notes: Before changing the value of this property, you must call lockForConfiguration: to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using key-value observing.
See "AVCaptureExposureMode" for possible values.
(Read and Write property)

4.35.35 exposurePointOfInterest as CGPointMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The point of interest for exposure.
Notes: Before changing the value of this property, you must call lockForConfiguration to acquire exclusive

access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.
(Read and Write property)

4.35.36 flashMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The current flash mode.
Notes: Before changing the value of this property, you must call lockForConfiguration: to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using key-value observing.

See "AVCaptureFlashMode" for possible values.
(Read and Write property)

4.35.37 focusMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The device's focus mode.
Notes: Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using key-value observing.

See "AVCaptureFocusMode" for possible values.
(Read and Write property)

4.35.38 focusPointOfInterest as CGPointMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The point of interest for focusing.
Notes: This property represents a CGPoint where { 0,0 } corresponds to the top left of the picture area,

and { 1,1 } corresponds to the bottom right in landscape mode with the home button on the right—this applies even if the device is in portrait mode.

Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.
(Read and Write property)

4.35.39 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.35.40 hasFlash as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the capture device has a flash. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.hasFlash)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.41 hasTorch as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value that specifies whether the capture device has a torch. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.hasTorch)
```

Notes: A torch is a light source, such as an LED flash, that is available on the device and used for illuminating captured content or providing general illumination. This property reflects whether the current device has such illumination hardware built-in.

Even if the device has a torch, that torch might not be available for use. Thus, you should also check the value of the torchAvailable property before using it.

You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.42 isAdjustingExposure as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The transport type of the receiver. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isAdjustingExposure)
```

Notes: The value of this property represents the transport type of the device (USB, PCI, etc). Transport types are defined in <IOKit/audio/IOAudioTypes.h>as kIOAudioDeviceTransportType*.
(Read only property)

4.35.43 isAdjustingFocus as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the device is currently adjusting its focus setting. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isAdjustingFocus)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.44 isAdjustingWhiteBalance as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the devise is currently adjusting the white balance. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isAdjustingWhiteBalance)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.45 isConnected as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the device is currently connected. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isConnected)
```

Notes: The value of this property indicates whether the device represented by the receiver is connected and available for use as a capture device. When the value of this property becomes false for a given instance, however, it will not become true again. If the same physical device again becomes available to the system, it will be represented using a new instance of AVCaptureDevice.

You can observe the value of this property using key-value observing to be notified when a device is no longer available.
(Read only property)

4.35.46 isExposurePointOfInterestSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the device supports a point of interest for exposure. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
```

```
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isExposurePointOfInterestSupported)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.47 isFocusPointOfInterestSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the device supports a point of interest for focus. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isFocusPointOfInterestSupported)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.48 isInUseByAnotherApplication as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the device is in use by another application. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isInUseByAnotherApplication)
```

Notes: You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.49 isRampingVideoZoom as Boolean

Plugin Version: 22.2, Platform: iOS, Targets: iOS only.

Function: A Boolean value that indicates whether a zoom transition is in progress.

Notes: (Read only property)

4.35.50 isSuspended as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the device is suspended. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.isSuspended)
```

Notes: Some devices disallow data capture due to a feature on the device.

For example, isSuspended returns true for an external iSight camera when its privacy iris is closed, or for the internal iSight camera on a notebook when the notebook's display is closed.

You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.51 localizedName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A localized human-readable name for the receiver. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeAudio)
MsgBox device.localizedName
```

Notes: You can use this property to display the name of a capture device in a user interface.
(Read only property)

4.35.52 manufacturer as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The human-readable manufacturer name for the receiver.
Notes: This property can be used to identify capture devices from a particular manufacturer. All Apple devices return "Apple Inc.". Devices from third party manufacturers may return an empty string.
Available on Mac OS X 10.9 or later.
(Read only property)

4.35.53 maxAvailableVideoZoomFactor as Double

Plugin Version: 22.2, Platform: iOS, Targets: iOS only.

Function: The maximum zoom factor allowed in the current capture configuration.
Notes: On single-camera devices, this value is always equal to the device format,Äôs videoMaxZoomFactor value. On a dual-camera device, the allowed range of video zoom factors can change if the device is delivering depth data to one or more capture outputs.
Setting the videoZoomFactor property to (or calling the rampToVideoZoomFactor:withRate method with) a value greater than the device format,Äôs videoMaxZoomFactor value always raises an exception. Setting the video zoom factor to a value between the maximum available zoom factor and the device format,Äôs maximum clamps the zoom setting to the maximum available value.
(Read only property)

4.35.54 minAvailableVideoZoomFactor as Double

Plugin Version: 22.2, Platform: iOS, Targets: iOS only.

Function: The minimum zoom factor allowed in the current capture configuration.
Notes: On single-camera devices, this value is always 1.0. On a dual-camera device, the allowed range of video zoom factors can change if the device is delivering depth data to one or more capture outputs.

Setting the videoZoomFactor property to (or calling the rampToVideoZoomFactor method with) a value less than 1.0 always raises an exception. Setting the video zoom factor to a value between 1.0 and the minimum available zoom factor clamps the zoom setting to the minimum.
(Read only property)

4.35.55 modelID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The model ID of the device. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
```

```
MsgBox "modelID: "+device.modelID
```

Notes: The value of this property is an identifier unique to all devices of the same model. The value is persistent across device connections and disconnections, and across different systems. For example, the model ID of the camera built in to two identical iPhone models will be the same even though they are different physical devices.
(Read only property)

4.35.56 position as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the physical position of the device hardware on the system. (read-only)
Notes: You can observe changes to the value of this property using key-value observing.
See "AVCaptureDevicePosition" for possible values.
(Read only property)

4.35.57 reactionEffectGesturesEnabled as Boolean

Plugin Version: 23.5, Platform: macOS, Targets: All.

Function: A class property indicating whether gesture detection will trigger reaction effects on the video stream.
Example:

```
If AVCaptureDeviceMBS.reactionEffectGesturesEnabled Then
MessageBox "reaction effects enabled"
Else
MessageBox "reaction effects not enabled."
End If
```

Notes: Gesture detection will only run when the device's activeFormat.reactionEffectsSupported is also true, which will be reflected by canPerformReactionEffects.

This property changes to reflect the Gestures state in Control Center.
Clients can call performEffectForReaction independently of whether gesture detection is enabled, reaction effects from either source will be intermixed.

Requires macOS 14.0 or iOS 17.0.
You may check AVCaptureReactionEffectStateMBS.available for whether this is available.
(Read only property)

4.35.58 reactionEffectsEnabled as Boolean

Plugin Version: 23.5, Platform: macOS, Targets: All.

Function: A class property indicating whether the application is suitable for reaction effects, either by automatic gesture detection, or by calls to performEffectForReaction().
Example:

```
If AVCaptureDeviceMBS.reactionEffectsEnabled Then
MessageBox "reaction effects enabled"
Else
MessageBox "reaction effects not enabled."
End If
```

Notes: Reactions are only rendered when the device's activeFormat.reactionEffectsSupported is also true, which will be reflected by canPerformReactionEffects when the feature is both enabled and supported.

On macOS, Reaction Effects are enabled by default for all applications. On iOS, Reaction Effects are enabled by default for video conferencing applications (apps that use "voip" as one of their UIBackgroundModes). Non video conferencing applications may opt in for Reaction Effects by adding the following key to their Info.plist:

```
<key>NSCameraReactionEffectsEnabled</key>
<true/>
```

Requires macOS 14.0 or iOS 17.0.
You may check AVCaptureReactionEffectStateMBS.available for whether this is available.
(Read only property)

4.35.59 torchMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The current torch mode.
Notes: Setting the value of this property also sets the torch level to its maximum current value.

Before setting the value of this property, call the isTorchModeSupported method to make sure the device supports the desired mode. Setting the device to an unsupported torch mode results in the raising of an exception. For a list of possible values for this property, see "AVCaptureTorchMode."

Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using key-value observing.
(Read and Write property)

4.35.60 transportControlsPlaybackMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The current playback mode. (read-only)
Notes: This property is only valid for devices that support transport control.
You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.61 transportControlsSpeed as Double

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The current playback speed. (read-only)
Notes: For devices that support transport control, the value of this property indicates the current playback speed of the deck. The following table gives examples of the meaning of values:

Value	Meaning
0.0	Stopped
1.0	Forward at normal speed.
-1.0	Reverse at normal speed.
2.0	Forward at 2x normal speed.

You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.62 transportControlsSupported as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the device supports transport control commands. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
MsgBox str(device.transportControlsSupported)
```

Notes: For devices with transport controls, such as AVC tape-based camcorders or pro capture devices with RS422 deck control, the value of this property is true. If transport controls are not supported, none of the associated transport control methods and properties are available on the receiver.

You can observe changes to the value of this property using key-value observing.
(Read only property)

4.35.63 transportType as Integer

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The transport type of the receiver. (read-only)
Notes: The value of this property represents the transport type of the device (USB, PCI, etc). Transport types are defined in <IOKit/audio/IOAudioTypes.h>as kIOAudioDeviceTransportType*.
(Read only property)

4.35.64 uniqueID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An ID unique to the model of device corresponding to the receiver. (read-only)
Example:

```
dim device as AVCaptureDeviceMBS
device = AVCaptureDeviceMBS.defaultDeviceWithMediaType(AVFoundationMBS.AVMediaTypeVideo)
dim u as string = device.uniqueID
MsgBox "uniqueID: "+u
```

Notes: Every available capture device has a unique ID that persists on one system across device connections and disconnections, application restarts, and reboots of the system itself. You can store the value returned by this property to recall or track the status of a specific device in the future.
(Read only property)

4.35.65 videoZoomFactor as Double

Plugin Version: 22.2, Platform: iOS, Targets: iOS only.

Function: A value that controls the cropping and enlargement of images captured by the device.
Notes: This value is a multiplier. For example, a value of 2.0 doubles the size of an image,Äôs subject

(and halves the field of view). Allowed values range from 1.0 (full field of view) to the value of the active format,Äôs videoMaxZoomFactor property. Setting the value of this property jumps immediately to the new zoom factor. For a smooth transition, use the rampToVideoZoomFactor method.
(Read and Write property)

4.35.66 whiteBalanceMode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The current white balance mode.
Notes: Before changing the value of this property, you must call lockForConfiguration to acquire exclusive access to the device's configuration properties. If you do not, setting the value of this property raises an exception. When you are done configuring the device, call unlockForConfiguration to release the lock and allow other devices to configure the settings.

You can observe changes to the value of this property using key-value observing.
See "AVCaptureWhiteBalanceMode" for possible values.
(Read and Write property)

4.35.67 Constants

Authorization

Constant	Value	Description
AVAuthorizationStatusAuthorized	3	The client is authorized to access the hardware supporting a media type is default for MacOS 10.13 or older.
AVAuthorizationStatusDenied	2	The user explicitly denied access to the hardware supporting a media ty the client.
AVAuthorizationStatusNotDetermined	0	Indicates that the user has not yet made a choice regarding whether the can access the hardware.
AVAuthorizationStatusRestricted	1	The client is not authorized to access the hardware for the media type user cannot change the client's status, possibly due to active restriction as parental controls being in place.

Device Position Modes

Constant	Value	Description
AVCaptureDevicePositionBack	1	The capture device is on the back of the unit.
AVCaptureDevicePositionFront	2	The capture device is on the front of the unit.
AVCaptureDevicePositionUnspecified	0	The capture device's position relative to the system hardware is unspecifi

Transport Control Modes

Constant	Value	Description
AVCaptureDeviceTransportControlsNotPlayingMode	0	Indicates that the tape transport is not threaded through
AVCaptureDeviceTransportControlsPlayingMode	1	Indicates that the tape transport is threaded through the

Exposure Mode Constants

Constant	Value	Description
AVCaptureExposureModeAutoExpose	1	The device continuously monitors exposure levels and auto essary.
AVCaptureExposureModeContinuousAutoExposure	2	The device performs an auto-expose operation now.
AVCaptureExposureModeLocked	0	The exposure setting is locked.

Flash Modes

Constant	Value	Description
AVCaptureFlashModeAuto	2	The capture device continuously monitors light levels and uses the flash when necessary.
AVCaptureFlashModeOff	0	The capture device flash is always off.
AVCaptureFlashModeOn	1	The capture device flash is always on.

Focus Modes

Constant	Value	Description
AVCaptureFocusModeAutoFocus	1	The capture device performs an autofocus operation now.
AVCaptureFocusModeContinuousAutoFocus	2	The capture device continuously monitors focus and auto focuses sary.
AVCaptureFocusModeLocked	0	The focus is locked.

Torch Modes

Constant	Value	Description
AVCaptureTorchModeAuto	2	The capture device continuously monitors light levels and uses the torch when necessary.
AVCaptureTorchModeOff	0	The capture device torch is always off.
AVCaptureTorchModeOn	1	The capture device torch is always on.

White Balance Modes

Constant	Value	Description
AVCaptureWhiteBalanceModeAutoWhiteBalance	1	The device performs an auto white balance opera
AVCaptureWhiteBalanceModeContinuousAutoWhiteBalance	2	The device continuously monitors white balance a
AVCaptureWhiteBalanceModeLocked	0	The white balance setting is locked.

4.36 class AVCaptureFileOutputMBS

4.36.1 class AVCaptureFileOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureFileOutput is an abstract sub-class of AVCaptureOutput that describes a file output destination to an AVCaptureSession.
Notes: For example, you use an instance of its concrete subclass, AVCaptureMovieFileOutput, to save capture output to a QuickTime movie file. The concrete subclasses of AVCaptureFileOutput are AVCaptureMovieFileOutput, which records media to a QuickTime movie file, and AVCaptureAudioFileOutput, which writes audio media to a variety of audio file formats.

This abstract superclass defines the interface for outputs that record media samples to files. File outputs can start recording to a new file using startRecordingToOutputFileURL method.

In OS X, on successive invocations of this method the output file can be changed dynamically without losing media samples. A file output can stop recording using the stopRecording method. Because files are recorded in the background, you need to specify a delegate for each new file to be notified when recorded files are finished.

In OS X, you can also set a delegate on the file output itself that can be used to control recording along exact media sample boundaries using the captureOutputDidOutputSampleBuffer event in AVFoundationMBS.
Subclass of the AVCaptureOutputMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.36.2 Methods

4.36.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.36.4 EnableEvents

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables the events for this output.
Notes: Those are normally handled automatically, but if you add event with AddHandler you need to call this method.

4.36.5 isRecording as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether recording is in progress.
Notes: The value of this property is true when the file output currently has a file to which it is writing new samples, false otherwise.

4.36.6 isRecordingPaused as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether recording to the current output file is paused.
Notes: This property indicates recording to the file returned by outputFileURL has been previously paused using the pauseRecording method. When a recording is paused, captured samples are not written to the output file, but new samples can be written to the same file in the future by calling resumeRecording.

4.36.7 outputFileURL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL to which output is directed. (read-only)

4.36.8 pauseRecording

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Pauses recording to the current output file.
Notes: This method causes the receiver to stop writing captured samples to the current output file returned by outputFileURL, but leaves the file open so that samples can be written to it in the future, if resumeRecording is called. This allows you to record multiple media segments that are not contiguous in time to a single file.

In OS X, if this method is called within the AVFoundationMBS captureOutputDidFinishRecordingToOutputFileAtURL event, the last samples written to the current file are guaranteed to be those that were output immediately before those in the sample buffer passed to that method.

4.36.9 recordedDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the duration of the media recorded to the current output file. (read-only)
Notes: If recording is in progress, this property returns the total time recorded so far.

4.36.10 recordedFileSize as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the size, in bytes, of the data recorded to the current output file. (read-only)
Notes: If a recording is in progress, this property returns the size in bytes of the data recorded so far.

4.36.11 resumeRecording

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Resumes recording to the current output file after it was previously paused using pauseRecording.
Notes: This method causes the receiver to resume writing captured samples to the current output file returned by outputFileURL, after recording was previously paused using pauseRecording. This allows you to record multiple media segments that are not contiguous in time to a single file.

In OS X, if this method is called within the AVFoundationMBS captureOutputDidFinishRecordingToOutputFileAtURL event, the first samples written to the current file are guaranteed to be those contained in the sample buffer passed to that method.

4.36.12 startRecordingToOutputFile(file as folderitem)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Starts recording to a given file.
Notes: Same as startRecordingToOutputFileURL, but with folderitem.

4.36.13 startRecordingToOutputFileURL(URL as string)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Starts recording to a given URL.
Notes: URL: The URL of the output file. This method throws an NSInvalidArgumentException if the URL is not a valid file URL.

The method sets the file URL to which the receiver is currently writing output media. If a file at the given

URL already exists when capturing starts, recording to the new file will fail.

In OS X, you do not need to call stopRecording before calling this method while another recording is in progress. If this method is invoked while an existing output file was already being recorded, no media samples will be discarded between the old file and the new file.

In iOS, this frame accurate file switching is not supported. You must call stopRecording before calling this method again to avoid any errors.

When recording is stopped either by calling stopRecording, by changing files using this method, or because of an error, the remaining data that needs to be included to the file will be written in the background. Therefore, you must specify a delegate that will be notified when all data has been written to the file using the AVFoundationMBS captureOutputDidFinishRecordingToOutputFileAtURL event. The recording delegate can also optionally implement methods that inform it when data starts being written, when recording is paused and resumed, and when recording is about to be finished.

In OS X, if this method is called within the AVFoundationMBS captureOutputDidOutputSampleBuffer event, the first samples written to the new file are guaranteed to be those contained in the sample buffer passed to that method.

Note: AVCaptureAudioFileOutput does not support startRecordingToOutputFileURL without filetype.

4.36.14 stopRecording

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the receiver to stop recording to the current file.
Notes: You can call this method when they want to stop recording new samples to the current file, and do not want to continue recording to another file. If you want to switch from one file to another, you should not call this method. Instead you should simply call startRecordingToOutputFileURL with the new file URL.

When recording is stopped either by calling this method, by changing files using startRecordingToOutputFileURL, or because of an error, the remaining data that needs to be included to the file will be written in the background. Therefore, before using the file, you must wait until the delegate that was specified in startRecordingToOutputFileURL is notified when all data has been written to the file using the AVFoundationMBS.captureOutputDidFinishRecordingToOutputFileAtURL method.

In OS X, if this method is called within the AVFoundationMBS captureOutputDidOutputSampleBuffer event, the last samples written to the current file are guaranteed to be those that were output immediately before those in the sample buffer passed to that method.

4.36.15 Properties

4.36.16 maxRecordedDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The longest duration allowed for the recording.
Notes: This property specifies a hard limit on the duration of recorded files. Recording is stopped when the limit is reached and the AVFoundationMBS.captureOutputDidFinishRecordingToOutputFileAtURL event is invoked with an appropriate error. The default value of this property is kCMTimeInvalid, which indicates no limit.
(Read and Write computed property)

4.36.17 maxRecordedFileSize as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The maximum size, in bytes, of the data that should be recorded by the receiver.
Notes: This property specifies a hard limit on the data size of recorded files. Recording is stopped when the limit is reached and the AVFoundationMBS captureOutputDidFinishRecordingToOutputFileAtURL event is invoked with an appropriate error. The default value of this property is 0, which indicates no limit.
(Read and Write computed property)

4.36.18 minFreeDiskSpaceLimit as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The minimum amount of free space, in bytes, required for recording to continue on a given volume.
Notes: This property specifies a hard lower limit on the amount of free space that must remain on a target volume for recording to continue. Recording is stopped when the limit is reached and the AVFoundationMBS captureOutputDidFinishRecordingToOutputFileAtURL event is invoked with an appropriate error.
(Read and Write computed property)

4.37 class AVCaptureInputMBS

4.37.1 class AVCaptureInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureInput is an abstract base-class describing an input data source to an AVCaptureSession object.
Notes: To associate an AVCaptureInput object with a session, call addInput on the session.

AVCaptureInput objects have one or more ports (instances of AVCaptureInputPort), one for each data stream they can produce. For example, an AVCaptureDevice object presenting one video data stream has one port.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.37.2 Methods

4.37.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.37.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.37.5 ports as AVCaptureInputPortMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture input's ports. (read-only)
Notes: The array contains one or more instances of AVCaptureInputPort.

Each individual AVCaptureInputPort instance posts an AVCaptureInputPortFormatDescriptionDidChangeNotification when the formatDescription of that port changes.

4.37.6 portWithMediaType(mediaType as string) as AVCaptureInputPortMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Finds the port in the ports array with given mediatype.
Notes: Returns nil if not found.

4.37.7 Properties

4.37.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.38 class AVCaptureInputPortMBS

4.38.1 class AVCaptureInputPortMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVCaptureInputPort represents a stream of data from a capture input.
Notes: Instances of AVCaptureInput have one or more input ports, one for each data stream they can produce. For example, an AVCaptureDeviceInput presenting one video data stream has one port.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.38.2 Methods

4.38.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.38.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.38.5 formatDescription as CMFormatDescriptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A description of the port format. (read-only)

4.38.6 input as AVCaptureInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The port's input. (read-only)

4.38.7 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The port's media type. (read-only)

4.38.8 Properties

4.38.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.38.10 Enabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the port is enabled.
Notes: (Read and Write computed property)

4.39 class AVCaptureMetadataOutputMBS

4.39.1 class AVCaptureMetadataOutputMBS

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: A capture output for processing timed metadata produced by a capture session.
Notes: Available for iOS only!

An AVCaptureMetadataOutput object intercepts metadata objects emitted by its associated capture connection and forwards them to a delegate object for processing. You can use instances of this class to process specific types of metadata included with the input data. You use this class the way you do other output objects, typically by adding it as an output to an AVCaptureSessionMBS object.
Subclass of the AVCaptureOutputMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 21.2
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.2
- MBS Xojo Plugins, version 21.2pr7

4.39.2 Methods

4.39.3 availableMetadataObjectTypes as string()

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: An array of strings identifying the types of metadata objects that can be captured.
Notes: Each string in the array corresponds to a possible value in the type property of the AVMetadataObjectMBS objects reported by the receiver. The available types are dependent on the capabilities of the AVCaptureInputPortMBS to which the receiver‚Äôs connection is attached.

4.39.4 Constructor

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: The constructor.

4.39.5 EnableEvents

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: Enables events, if you use addHandler to connect them.

4.39.6 metadataObjectTypes as String()

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: An array of strings identifying the types of metadata objects to process.
Notes: This property is used to filter the metadata objects reported by the receiver. Only metadata objects whose type matches one of the strings in this property are forwarded to the captureOutput:didOutputMetadataObjects event for processing.

When assigning a new array to this property, each of the type strings must be present in the array returned by the availableMetadataObjectTypes property; otherwise, the receiver raises an NSExceptionMBS.
The default is an empty array, and as a result, no metadata objects are forwarded to the event. The same result can be achieved by setting the property to nil. This default behavior maximizes both performance and battery life.

4.39.7 SetMetadataObjectTypes(items() as String = nil)

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: Sets the array of strings identifying the types of metadata objects to process.
Example:

```
Dim o As New AVCaptureMetadataOutputMBS
Dim types() As String

Dim availableTypes() As String = o.availableMetadataObjectTypes

// check if we can do EAN 13 and then add it
If availableTypes.IndexOf(AVMetadataObjectMBS.AVMetadataObjectTypeEAN13Code) >= 0 Then
types.append(AVMetadataObjectMBS.AVMetadataObjectTypeEAN13Code)
End If

o.SetMetadataObjectTypes types
```

Notes: This property is used to filter the metadata objects reported by the receiver. Only metadata objects whose type matches one of the strings in this property are forwarded to the captureOutput:didOutputMetadataObjects event for processing.

When assigning a new array to this property, each of the type strings must be present in the array returned by the availableMetadataObjectTypes property; otherwise, the receiver raises an NSExceptionMBS.

The default is an empty array, and as a result, no metadata objects are forwarded to the event. The same result can be achieved by setting the property to nil. This default behavior maximizes both performance and battery life.

4.39.8 Properties

4.39.9 rectOfInterest as CGRectMBS

Plugin Version: 21.2, Platform: iOS, Targets: iOS only.

Function: A rectangle of interest for limiting the search area for visual metadata.
Notes: The value of this property is a CGRect value that determines the object,Äôs rectangle of interest for each frame of video.
The rectangle's origin is top left and is relative to the coordinate space of the device providing the metadata.
Specifying a rectangle of interest may improve detection performance for certain types of metadata. Metadata objects whose bounds do not intersect with the rectOfInterest will not be returned.
The default value of this property is a rectangle of (0.0, 0.0, 1.0, 1.0).
(Read and Write property)

4.40 class AVCaptureMovieFileOutputMBS

4.40.1 class AVCaptureMovieFileOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureMovieFileOutput is a concrete sub-class of AVCaptureFileOutput you use to capture data to a QuickTime movie.
Notes: The timeMapping.target.start of the first track segment must be kCMTimeZero, and the timeMapping.target.start of each subsequent track segment must equal CMTimeRangeGetEnd(<#the previous AVCompositionTrackSegment's timeMapping.target#>). You can use validateTrackSegments to ensure that an array of track segments conforms to this rule.
Subclass of the AVCaptureFileOutputMBS class.

4.40.2 Methods

4.40.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.40.4 EnableEvents

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables the events for this output.
Notes: Those are normally handled automatically, but if you add event with AddHandler you need to call this method.

4.40.5 metadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The metadata for the output file.
Notes: The array contains AVMetadataItem objects. You use this array to add metadata such as copyright, creation date, and so on, to the recorded movie file.

4.40.6 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the metadata for the output file.
Notes: The array contains AVMetadataItem objects. You use this array to add metadata such as copyright, creation date, and so on, to the recorded movie file.

4.40.7 Properties

4.40.8 movieFragmentInterval as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the number of seconds of output that are written per fragment.
Notes: The default is 10 seconds. Set to kCMTimeInvalid to disable movie fragment writing (not typically recommended).

A QuickTime movie is comprised of media samples and a sample table identifying their location in the file. A movie file without a sample table is unreadable.

In a processed file, the sample table typically appears at the beginning of the file. It may also appear at the end of the file, in which case the header contains a pointer to the sample table at the end. When a new movie file is being recorded, it is not possible to write the sample table since the size of the file is not yet known. Instead, the table is must be written when recording is complete. If no other action is taken, this means that if the recording does not complete successfully (for example, in the event of a crash), the file data is unusable (because there is no sample table). By periodically inserting "movie fragments" into the movie file, the sample table can be built up incrementally. This means that if the file is not written completely, the movie file is still usable (up to the point where the last fragment was written).
(Read and Write computed property)

4.40.9 outputSettingsForConnection(connection as AVCaptureConnectionMBS) as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the options the receiver uses to re-encode media from the given connection as it is being recorded.
Notes: connection: The connection delivering the media to be re-encoded.
Returns an Dictionary of output settings.

See AVAudioSettings.h for audio connections or AVVideoSettings.h for video connections for more information on how to construct an output settings dictionary. If the returned value is an empty dictionary (i.e.

new dictionary, the format of the media from the connection will not be changed before being written to the file. If setOutputSettings was called with a nil dictionary, this method returns a non-nil dictionary reflecting the settings used by the AVCaptureSession's current sessionPreset.
(Read and Write computed property)

4.41 class AVCaptureOutputMBS

4.41.1 class AVCaptureOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureOutput is an abstract base-class describing an output destination of an AVCaptureSession object.
Notes: AVCaptureOutput provides an abstract interface for connecting capture output destinations, such as files and video previews, to an capture session (an instance of AVCaptureSession). A capture output can have multiple connections represented by AVCaptureConnection objects, one for each stream of media that it receives from a capture input (an instance of AVCaptureInput). A capture output does not have any connections when it is first created. When you add an output to a capture session, connections are created that map media data from that session's inputs to its outputs.

You can add concrete AVCaptureOutput instances to an capture session using addOutput.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.41.2 Methods

4.41.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.41.4 connections as AVCaptureConnectionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture output object's connections. (read-only)
Notes: The value of this property is an array of AVCaptureConnection objects, each describing the mapping between the receiver and the capture input ports (see AVCaptureInputPort) of one or more capture inputs (see AVCaptureInput).

4.41.5 connectionWithMediaType(mediaType as string) as AVCaptureConnectionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the first connection in the connections array with an input port of a specified media type.
Notes: mediaType: An AVMediaType constant from AVMediaFormat.h, for example, AVMediaTypeVideo.

Returns the first capture connection in the connections array that has an AVCaptureInputPort with media type mediaType, or nil if no connection with the specified media type is found.

4.41.6 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.41.7 Properties

4.41.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.42 class AVCaptureReactionEffectStateMBS

4.42.1 class AVCaptureReactionEffectStateMBS

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Reports the state of a reaction performed on an AVCaptureDeviceMBS.
Example:

```
dim device as AVCaptureDeviceMBS // your device

dim effects() as AVCaptureReactionEffectStateMBS = device.reactionEffectsInProgress

for each effect as AVCaptureReactionEffectStateMBS in effects
MessageBox effect.reactionType
next
```

Notes: AVCaptureReactionEffectStateMBS may be obtained by calling reactionEffectsInProgress in AVCaptureDeviceMBS. When canPerformReactionEffects in AVCaptureDeviceMBS returns true, new entries are added either by calling performReactionEffect, or by gesture detection in the capture stream when AVCaptureDevice.reactionEffectGesturesEnabled. The effect rendering is done before frames are given to the capture client, and these status objects let you know when these effects are performed.

Requires macOS 14.0 or iOS 17.0.
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 23.5
- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5
- MBS Xojo Plugins, version 23.5pr6

4.42.2 Methods

4.42.3 available as boolean

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Whether this class is available.
Example:

```
If AVCaptureReactionEffectStateMBS.available Then
MessageBox "available"
Else
MessageBox "not available."
```

End If

Notes: Returns true on macOS 14.0 or iOS 17.0.

4.42.4 AVCaptureReactionTypeBalloons as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features balloons rising through the scene.

4.42.5 AVCaptureReactionTypeConfetti as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features festive spots of color falling through the scene.

4.42.6 AVCaptureReactionTypeFireworks as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features fireworks bursting in the background.

4.42.7 AVCaptureReactionTypeHeart as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features one or more heart symbols.

4.42.8 AVCaptureReactionTypeLasers as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features a bright laser display projecting into the scene.

4.42.9 AVCaptureReactionTypeRain as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features a dark and stormy night.

4.42.10 AVCaptureReactionTypeThumbsDown as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features a thumbs-down symbol.

4.42.11 AVCaptureReactionTypeThumbsUp as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: One of the rection types that can be performed in an effect.
Notes: Indicates a reaction which features a thumbs-up symbol.

4.42.12 Constructor

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: The private constructor.

4.42.13 ReactionSystemImageNameForType(reactionType as String) as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Returns the name for NSImage systemImageNamed method to obtain the recommended iconography for a specified reaction type.
Example:

```
// loads image into canvas
dim effect as string = AVCaptureReactionEffectStateMBS.AVCaptureReactionTypeBalloons
dim name as string = AVCaptureReactionEffectStateMBS.ReactionSystemImageNameForType(effect)
dim image as NSImageMBS = NSImageMBS.imageWithSystemSymbolName(name)

// change size to match
image.setSize canvas1.Width, canvas1.Height
```

```
// and render picture
canvas1.Backdrop = image.CopyPictureWithAlpha
```

4.42.14 Properties

4.42.15 endTime as CMTimeMBS

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Provides the presentation time of the frame following the last frame where the effect is seen.
Notes: Will be CMTimeMBS.kCMTimeInvalid while the effect is in progress, but will be updated to a valid time when the reaction effect completes and the AVCaptureReactionEffectState is removed from reactionEffectsInProgress array.
(Read only property)

4.42.16 Handle as Integer

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

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

4.42.17 reactionType as String

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Indicates the reaction which is running.
Notes: There may be multiple reactions of the same type at a given time.
Some may come from gesture detection, some may come from calls to performReactionEffect method in AVCaptureDeviceMBS class.
(Read only property)

4.42.18 startTime as CMTimeMBS

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: All.

Function: Provides the presentation time of the first frame where the effect is being rendered.
Notes: (Read only property)

4.43 class AVCaptureScreenInputMBS

4.43.1 class AVCaptureScreenInputMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: AVCaptureScreenInput is a concrete subclass of AVCaptureInput that provides an interface for capturing media from a screen or a portion of a screen.
Notes: Instances of AVCaptureScreenInput are input sources for AVCaptureSession objects that provide media data from one of the screens connected to the system, represented by CGDirectDisplayIDs.
Subclass of the AVCaptureInputMBS class.
Blog Entries

- MBS Xojo Plugins, version 18.4pr1

4.43.2 Methods

4.43.3 Constructor(CGDisplay as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a capture screen input that provides media data from a given display.
Notes: CGDisplay: The display from which to capture video. Must be CGDisplayMBS object.
Handle is non zero on success.

4.43.4 Properties

4.43.5 capturesCursor as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A property indicating whether the cursor should be rendered to the captured output.
Notes: By default, AVCaptureScreenInput draws the cursor in its captured output. If this property is set to false, the captured output contains only the windows on the screen. Cursor is omitted. Note that cursor position and mouse button state at the time of capture is preserved in CMSampleBuffers emitted from AVCaptureScreenInput. See the inline documentation for kCMIOSampleBufferAttachmentKey_MouseAndKeyboardModifiers in <CoreMediaIO/CMIOSampleBuffer.h>

Available in Mac OS X 10.8 and newer.
(Read and Write computed property)

4.43.6 capturesMouseClicks as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether mouse clicks should be highlighted in the captured output.
Notes: By default, AVCaptureScreenInput does not highlight mouse clicks in its captured output.

If you set this property is set to true, mouse clicks are highlighted (a circle is drawn around the mouse for the duration of the click) in the captured output.
(Read and Write computed property)

4.43.7 cropRect as CGRectMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the bounding rectangle of the screen area to be captured, in pixels.
Notes: By default, AVCaptureScreenInput captures the entire area of the displayID with which it is associated.

Set the value of this property to limit the capture rectangle to a subsection of the screen.

The rectangle should define a smaller section of the screen in the screen's coordinate system. The origin (0,0) is the bottom-left corner of the screen.
(Read and Write computed property)

4.43.8 minFrameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The screen input's minimum frame duration.
Notes: The minFrameDuration is the reciprocal of its maximum frame rate.

You use this property to request a maximum frame rate at which the input produces video frames. The requested rate may not be achievable due to overall bandwidth, so actual frame rates may be lower.
(Read and Write computed property)

4.43.9 removesDuplicateFrames as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A property indicating whether duplicate frames should be removed by the input.

Notes: By default, AVCaptureScreenInput performs frame differencing and when it detects duplicate frames, it drops them. If this property is set to false, the captured output receives all frames from the input.

Available in Mac OS X 10.8 and newer.
(Read and Write computed property)

4.43.10 scaleFactor as Double

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the factor by which video buffers captured from the screen are to be scaled.
Notes: By default, AVCaptureScreenInput captures the video buffers from the display at a scale factor of 1.0 (no scaling). Set this property to scale the buffers by a given factor; for example a 320x240 capture area with a scaleFactor of 2.0 produces video buffers at 640x480.
(Read and Write computed property)

4.44 class AVCaptureSessionMBS

4.44.1 class AVCaptureSessionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVCaptureConnection object represents a connection between capture input and capture output objects associated with a capture session.
Notes: Capture inputs (instances of AVCaptureInput) have one or more input ports (instances of AVCaptureInputPort). Capture outputs (instances of AVCaptureOutput) can accept data from one or more sources (for example, an AVCaptureMovieFileOutput object accepts both video and audio data).

You can only add an AVCaptureConnection instance to a session using addConnection if canAddConnection returns true. When using addInput or addOutput, connections are formed automatically between all compatible inputs and outputs. You only need to add connections manually when adding an input or output with no connections. You can also use connections to enable or disable the flow of data from a given input or to a given output.

4.44.2 Methods

4.44.3 addConnection(connection as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds a given capture connection to the session.
Notes: connection: The capture connection to add to the session.

You can only add an AVCaptureConnection instance to a session using this method if canAddConnection returns true.
When using addInput or addOutput, connections are formed automatically between all compatible inputs and outputs. Manually adding connections is only necessary when adding an input or output with no connections.

4.44.4 addInput(connection as AVCaptureInputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds a given input to the session.
Notes: input: An input to add to the session.

You can only add an input to a session using this method if canAddInput returns true.
You can invoke this method while the session is running.

4.44.5 addInputWithNoConnections(input as AVCaptureInputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds an capture input to the session without forming any connections.
Notes: input: The capture input to add to the session.
You can invoke this method while the session is running.
Typically you should use addInput to add an input to a session. You use this method if you need fine-grained control over which inputs are connected to which outputs.

4.44.6 addOutput(connection as AVCaptureOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds a given output to the session.
Notes: output: An output to add to the session.
You can only add an output to a session using this method if canAddOutput returns true.
You can invoke this method while the session is running.

4.44.7 addOutputWithNoConnections(output as AVCaptureOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds an capture output to the session without forming any connections.
Notes: output: The capture output to add to the session.

You can invoke this method while the session is running.
Typically you should use addOutput to add an output to a session. You use this method if you need fine-grained control over which inputs are connected to which outputs.

4.44.8 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.44.9 beginConfiguration

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the start of a set of configuration changes to be made atomically.
Notes: You use beginConfiguration and commitConfiguration to batch multiple configuration operations on a running session into an atomic update.

After calling beginConfiguration, you can for example add or remove outputs, alter the sessionPreset, or configure individual capture input or output properties. No changes are actually made until you invoke commitConfiguration, at which time they are applied together.

4.44.10 canAddConnection(connection as AVCaptureConnectionMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given connection can be added to the receiver.
Notes: connection: An AVCaptureConnection instance.
Returns true if connection can be added to the receiver, otherwise false.

4.44.11 canAddInput(input as AVCaptureInputMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given input can be added to the session.
Notes: input: An input that you want to add to the session.
Returns true if input can be added to the session, otherwise false.

4.44.12 canAddOutput(input as AVCaptureOutputMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given output can be added to the session.
Notes: output: An output that you want to add to the session.
Returns true if output can be added to the session, otherwise false.

4.44.13 canSetSessionPreset(preset as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver can use the given preset.
Notes: preset: A preset you would like to set for the receiver. For possible values, see AVFoundationMBS. Returns true if the receiver can use preset, otherwise false.

4.44.14 commitConfiguration

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Commits a set of configuration changes.
Notes: For discussion, see beginConfiguration.

4.44.15 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.44.16 inputs as AVCaptureInputMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture session's inputs. (read-only)
Notes: The array contains instances of subclasses of AVCaptureInput.

4.44.17 isRunning as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the receiver is running. (read-only)
Notes: You can observe the value of this property using key-value observing.

4.44.18 outputs as AVCaptureOutputMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture session's outputs. (read-only)
Notes: The array contains instances of subclasses of AVCaptureOutput.

4.44.19 removeConnection(connection as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a capture connection from the session.
Notes: connection: The capture connection to remove from the session.
You can invoke this method while the session is running.

4.44.20 removeInput(connection as AVCaptureInputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a given input.
Notes: input: An input to remove from the receiver.
You can invoke this method while the session is running.

4.44.21 removeOutput(connection as AVCaptureOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a given output.
Notes: output: An output to remove from the receiver.
You can invoke this method while the session is running.

4.44.22 startRunning

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the receiver to start running.
Notes: startRunning and stopRunning are asynchronous operations. If an error occurs occur during a capture session, you receive an AVCaptureSessionRuntimeErrorNotification.

4.44.23 stopRunning

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the receiver to stop running.
Notes: startRunning and stopRunning are asynchronous operations. If an error occurs occur during a capture session, you receive an AVCaptureSessionRuntimeErrorNotification.

4.44.24 Properties

4.44.25 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.44.26 sessionPreset as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A constant value indicating the quality level or bitrate of the output.
Notes: You use this property to customize the quality level or bitrate of the output. For possible values of sessionPreset, see AVFoundationMBS. The default value is AVCaptureSessionPresetHigh.

You can set this value while the session is running.
You can only set a preset if canSetSessionPreset: returns true for that preset.
(Read and Write computed property)

4.45 class AVCaptureStillImageOutputMBS

4.45.1 class AVCaptureStillImageOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureStillImageOutput is a concrete sub-class of AVCaptureOutput that you use to capture a high-quality still image with accompanying metadata.
Notes: Subclass of the AVCaptureOutputMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.4pr6

4.45.2 Methods

4.45.3 availableImageDataCodecTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The supported image codec formats that can be specified in outputSettings. (read-only)
Example:

```
dim a as new AVCaptureStillImageOutputMBS
MsgBox join(a.availableImageDataCodecTypes)
```

Notes: The value of this property is an array of Strings that you can use as values for the AVVideoCodecKey in the outputSettings property.

4.45.4 availableImageDataCVPixelFormatTypes as Integer()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The supported image pixel formats that can be specified in outputSettings. (read-only)
Example:

```
dim a as new AVCaptureStillImageOutputMBS
dim availableImageDataCVPixelFormatTypes() as Integer = a.availableImageDataCVPixelFormatTypes
break // check array in debugger
```

Notes: The value of this property is an array of numbers that you can use as values for the kCVPixelBufferPixelFormatTypeKey in the outputSettings property.
Currently only ”jpeg”.

Available in OS X v10.7 and later.

4.45.5 captureStillImageAsynchronously(connection as AVCaptureConnectionMBS, prepareJpegStillImage as boolean, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initiates a still image capture and returns immediately.
Notes: connection: The connection from which to capture the image.

Pass true for prepareJpegStillImage to have the plugin prepare jpeg data for you.

The buffer attachments may contain metadata appropriate to the image data format. For example, a buffer containing JPEG data may carry a kCGImagePropertyExifDictionary as an attachment. See ImageIO/CGImageProperties.h for a list of keys and value types.
error
If the request could not be completed, an NSError object that describes the problem; otherwise nil.

This method returns immediately after it is invoked, later calling the AVFoundationMBS.captureStillImageAsynchronouslyCompleted event when image data is ready. If the request could not be completed, the error parameter will contain an NSError object describing the failure.

Available in OS X v10.7 and later.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.45.6 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.45.7 jpegStillImageNSDataRepresentation(jpegSampleBuffer as CMSampleBufferMBS) as memoryblock

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an memoryblock representation of a still image data and metadata attachments in a

JPEG sample buffer.
Notes: jpegSampleBuffer: The sample buffer carrying JPEG image data, optionally with Exif metadata sample buffer attachments.
This method throws an NSInvalidArgumentException if jpegSampleBuffer is nil or not in the JPEG format.

Returns memoryblock representation of jpegSampleBuffer.

This method merges the image data and Exif metadata sample buffer attachments without re-compressing the image.
The returned memoryblock object is suitable for writing to disk.

Available in OS X v10.7 and later.

4.45.8 Properties

4.45.9 isCapturingStillImage as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A boolean value that becomes true when a still image is being captured.
Notes: The value of this property is a BOOL that becomes true when a still image is being captured, and false when no still image capture is underway. This property is key-value observable.
(Read only property)

4.45.10 outputSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The compression settings for the output.
Notes: You specify the compression settings using keys from AVVideoSettings.h, or a dictionary of pixel buffer attributes using keys from CVPixelBuffer.h.

Currently the only supported keys are AVVideoCodecKey and kCVPixelBufferPixelFormatTypeKey. The recommended values are kCMVideoCodecType_JPEG, kCVPixelFormatType_420YpCbCr8BiPlanarFullRange and kCVPixelFormatType_32BGRA.

Available in OS X v10.7 and later.
(Read and Write computed property)

4.46 class AVCaptureVideoDataOutputMBS

4.46.1 class AVCaptureVideoDataOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureVideoDataOutput is a concrete sub-class of AVCaptureOutput you use to process uncompressed frames from the video being captured, or to access compressed frames.
Notes: An instance of AVCaptureVideoDataOutput produces video frames you can process using other media APIs. You can access the frames with the AVFoundationMBS.captureOutputDidOutputSampleBuffer event.
Subclass of the AVCaptureOutputMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.3pr3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.4
- MBS Xojo Plugins, version 18.4pr6
- MBS Xojo Plugins, version 18.4pr3
- AVCaptureVideoDataOutput improvements

Xojo Developer Magazine

- 17.1, page 11: News
- 16.6, page 9: News

4.46.2 Methods

4.46.3 availableVideoCodecTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the supported video codec formats that can be specified in videoSettings. (read-only)
Notes: The value of this property is an array of String objects you can use as values for the AVVideoCodecKey in the videoSettings property. The first format in the returned list is the most efficient output format.

4.46.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.46.5 Constructor(CIDetector as Variant) 350

4.46.5 Constructor(CIDetector as Variant)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The constructor.
Notes: This allows you to use a CIDetectorMBS object with AVCaptureVideoDataOutputMBS.
The plugin will run the detector in a helper thread, so you have more time on the main thread for your application and better performance.
See also:

- 4.46.4 Constructor 349

4.46.6 EnableEvents

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Enables the events for this output.
Notes: Those are normally handled automatically, but if you add event with AddHandler you need to call this method.

4.46.7 Properties

4.46.8 Detector as Variant

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The CIDetectorMBS object used here.
Notes: (Read only property)

4.46.9 PrepareCIImage as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Whether CIImageMBS should be created on preemptive thread.
Notes: If enabled, the plugin will do the work to create CIImage on the preemptive thread before calling AVFoundationMBS.captureOutputDidOutputSampleBuffer event on main thread. This may make your app more responsive on the main thread.
(Read and Write property)

4.46.10 PrepareJPEGData as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Whether JPEGData should be created on preemptive thread.
Notes: If enabled, the plugin will do the work to create JPEGData on the preemptive thread before calling AVFoundationMBS.captureOutputDidOutputSampleBuffer event on main thread. This may make your app more responsive on the main thread.
(Read and Write property)

4.46.11 PrepareNSImage as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Whether NSImageMBS should be created on preemptive thread.
Notes: If enabled, the plugin will do the work to create NSImage on the preemptive thread before calling AVFoundationMBS.captureOutputDidOutputSampleBuffer event on main thread. This may make your app more responsive on the main thread.
(Read and Write property)

4.46.12 alwaysDiscardsLateVideoFrames as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether video frames are dropped if they arrive late.
Notes: When the value of this property is true, the object immediately discards frames that are captured while the dispatch queue handling existing frames is blocked in the AVFoundationMBS.captureOutputDidOutputSampleBuffer event.

When the value of this property is false, delegates are allowed more time to process old frames before new frames are discarded, but application memory usage may increase significantly as a result.

The default is true.
(Read and Write computed property)

4.46.13 videoSettings as dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The compression settings for the output.
Notes: The dictionary contains values for compression settings keys defined in AVVideoSettings.h, or pixel buffer attributes keys defined in <CoreVideo/CVPixelBuffer.h>(see CVPixelBufferRef). The only key currently supported is the kCVPixelBufferPixelFormatTypeKey key.
(The plugin defines such keys in AVFoundationMBS)

To get possible values for the supported video pixel formats (kCVPixelBufferPixelFormatTypeKey) and video codec formats (AVVideoCodecKey), see availableVideoCVPixelFormatTypes and availableVideoCodecTypes respectively.

To receive samples in their device native format, set this property to nil:

If you set this property to nil and then subsequently query it, you will get a dictionary reflecting the settings used by the capture sessions's current sessionPreset.
(Read and Write computed property)

4.47 class AVCaptureVideoPreviewLayerMBS

4.47.1 class AVCaptureVideoPreviewLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVCaptureVideoPreviewLayer is a subclass of CALayer that you use to display video as it is being captured by an input device.
Notes: You use the videoGravity property to influence how content is viewed relative to the layer bounds. On some hardware configurations, you can manipulate the orientation of the layer using orientation and mirrored.
Subclass of the CALayerMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 22.2
- MBS Xojo Plugins, version 22.2pr7
- AVCaptureVideoDataOutput improvements

4.47.2 Methods

4.47.3 captureDevicePointOfInterestForPoint(pointInLayer as CGPointMBS) as CGPointMBS

Plugin Version: 22.2, Platform: macOS, Targets: All.

Function: Converts a point in layer coordinates to a point of interest in the coordinate space of the capture device providing input to the layer.
Notes: pointInLayer: A CGPoint in layer coordinates.

Returns a CGPoint in the coordinate space of the capture device providing input to the layer.

AVCaptureDevice pointOfInterest is expressed as a CGPoint where { 0,0 } represents the top left of the picture area, and { 1,1 } represents the bottom right on an unrotated picture. This convenience method converts a point in the coordinate space of the receiver to a point of interest in the coordinate space of the AVCaptureDevice providing input to the receiver. The conversion takes frameSize and videoGravity into consideration.

4.47.4 connection as AVCaptureConnectionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture connection describing the AVCaptureInputPort to which the preview layer is connected. (read-only)

Notes: If you invoke Constructor, layerWithSession, or set session with a valid AVCaptureSession instance, a connection is formed to the first eligible video AVCaptureInput object. If the preview layer is detached from a session, the connection property becomes nil.

4.47.5 Constructor(session as AVCaptureSessionMBS, WithConnection as boolean = true)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a preview layer with a given capture session.
Notes: Only pass WithConnection=false here if you intend to manually form a connection between a desired AVCaptureInputPort object and the receiver using addConnection.

4.47.6 layerWithSession(session as AVCaptureSessionMBS) as AVCaptureVideoPreviewLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a preview layer initialized with a given capture session.
Notes: session: The capture session from which to derive the preview.

Returns a preview layer initialized to use session.

4.47.7 layerWithSessionWithNoConnection(session as AVCaptureSessionMBS) as AVCaptureVideoPreviewLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a preview layer using a given capture session but without making any connections.
Notes: session: The capture session to be previewed.
Return a preview layer initialized with session but with no connections to any of the session's eligible video inputs

Only use this method if you intend to manually form a connection between a desired AVCaptureInputPort object and the receiver using addConnection.

4.47.8 metadataOutputRectOfInterestForRect(rectInLayerCoordinates as CGRectMBS) as CGRectMBS

Plugin Version: 22.2, Platform: macOS, Targets: All.

Function: Converts a rectangle in layer coordinates to a rectangle of interest in the coordinate space of an AVCaptureMetadataOutput whose capture device is providing input to the layer.
Notes: rectInLayerCoordinates: A CGRect in layer coordinates.

Returns a CGRect in the coordinate space of the metadata output whose capture device is providing input to the layer.

AVCaptureMetadataOutput rectOfInterest is expressed as a CGRect where { 0,0 } represents the top left of the picture area, and { 1,1 } represents the bottom right on an unrotated picture. This convenience method converts a rectangle in the coordinate space of the receiver to a rectangle of interest in the coordinate space of an AVCaptureMetadataOutput whose AVCaptureDevice is providing input to the receiver. The conversion takes frame size and videoGravity into consideration.

4.47.9 pointForCaptureDevicePointOfInterest(captureDevicePointOfInterest as CGPointMBS) as CGPointMBS

Plugin Version: 22.2, Platform: macOS, Targets: All.

Function: Converts a point of interest in the coordinate space of the capture device providing input to the layer to a point in layer coordinates.
Notes: captureDevicePointOfInterest: A CGPoint in the coordinate space of the capture device providing input to the layer.

Returns a CGPoint in layer coordinates.

AVCaptureDevice pointOfInterest is expressed as a CGPoint where { 0,0 } represents the top left of the picture area, and { 1,1 } represents the bottom right on an unrotated picture. This convenience method converts a point in the coordinate space of the AVCaptureDevice providing input to the coordinate space of the receiver. The conversion takes frame size and videoGravity into consideration.

4.47.10 rectForMetadataOutputRectOfInterest(rectInMetadataOutputCoordinates as CGRectMBS) as CGRectMBS

Plugin Version: 22.2, Platform: macOS, Targets: All.

Function: Converts a rectangle of interest in the coordinate space of an AVCaptureMetadataOutput whose capture device is providing input to the layer to a rectangle in layer coordinates.
Notes: rectInMetadataOutputCoordinates: A CGRect in the coordinate space of the metadata output

whose capture device is providing input to the layer.

Returns a CGRect in layer coordinates.

AVCaptureMetadataOutput rectOfInterest is expressed as a CGRect where { 0,0 } represents the top left of the picture area, and { 1,1 } represents the bottom right on an unrotated picture. This convenience method converts a rectangle in the coordinate space of an AVCaptureMetadataOutput whose AVCaptureDevice is providing input to the coordinate space of the receiver. The conversion takes frame size and videoGravity into consideration.

4.47.11 setSessionWithNoConnection(session as AVCaptureSessionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Attaches the receiver to a given session without implicitly forming a connection.
Notes: This method attaches the receiver to a given session without implicitly forming a connection to the first eligible video AVCaptureInputPort object. You should only use this method if you intend to manually form a connection between a desired AVCaptureInputPort object and the receiver using addConnection.

4.47.12 Properties

4.47.13 session as AVCaptureSessionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The capture session instance being previewed.
Notes: (Read and Write computed property)

4.47.14 videoGravity as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates how the video is displayed within a player layer's bounds rect.
Notes: Options are AVLayerVideoGravityResizeAspect, AVLayerVideoGravityResizeAspectFill and AVLayerVideoGravityResize. The default is AVLayerVideoGravityResizeAspect.

This property is animatable.
(Read and Write computed property)

4.48 control AVCaptureViewControlMBS

4.48.1 control AVCaptureViewControlMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The capture control for Mac.
Notes: AVCaptureView is a subclass of NSView that can be used to display standard user interface controls for capturing media data.
MBS Plugin wraps this as a control for Xojo.
Please notice this is 64-bit only and requires MacOS 10.9 or newer.
Blog Entries

- MBS Xojo Plugins, version 22.0pr8
- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- AVCaptureVideoDataOutput improvements
- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr2

Videos

- Presentation from Munich conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 16.5, page 9: News

4.48.2 Methods

4.48.3 Available as Boolean

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

4.48.4 setSession(session as AVCaptureSessionMBS, showVideoPreview as boolean, showAudioPreview as boolean)

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Sets the session represented by this view.
Notes: session: The session to be represented.
showVideoPreview: Whether or not video preview should be shown. If YES, capture inputs for video media data will be added, removed, or modified depending on device availability and user selection.
showAudioPreview: Whether or not audio preview should be shown. If YES, capture inputs for audio media data will be added, removed, or modified depending on device availability and user selection.

The view must either show audio preview or video preview or both. Furthermore, the view may modify the capture session, for example, to access media data for preview or when the user select a new capture source. Only the default session is started and stopped automatically. The provided session must be manually started and stopped.

4.48.5 Properties

4.48.6 controlsStyle as Integer

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The style of the capture controls pane associated with the view.
Notes: (Read and Write property)

4.48.7 fileOutput as AVCaptureFileOutputMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: A capture file output used to record media data.
Notes: The value of this property is the first instance of AVCaptureFileOutput contained in the session's outputs array or nil if no such instance is found. In the latter case the capture view's start recording button will be disabled. However, the controls for choosing input sources may still be enabled.
(Read only property)

4.48.8 session as AVCaptureSessionMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: A capture session represented by this view.
Notes: Modifying the capture session will impact its visual representation in the view. The default value is

a session configured for movie file recordings of audio and video media data. Use setSession() to change the value of this property.
(Read only property)

4.48.9 videoGravity as String

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: A string defining how the video is displayed within the views bounds rect.
Notes: Options are AVLayerVideoGravityResize, AVLayerVideoGravityResizeAspect and AVLayerVideoGravityResizeAspectFill. AVLayerVideoGravityResizeAspect is default. See <AVFoundation/AVAnimation.h>for a description of these options.
(Read and Write property)

4.48.10 View as NSViewMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The reference to the AVCaptureView object.
Notes: (Read only property)

4.48.11 Events

4.48.12 BoundsChanged

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.48.13 Close

Plugin Version: 18.3, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

4.48.14 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.

4.48.15 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 18.3, 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.

4.48.16 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.

4.48.17 EnableMenuItems

Plugin Version: 18.3, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

4.48.18 FrameChanged

Plugin Version: 18.3, 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.

4.48.19 GotFocus

Plugin Version: 18.3, 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.

4.48.20 LostFocus

Plugin Version: 18.3, 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.

4.48.21 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 18.3, 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.

4.48.22 MouseDrag(x as Integer, y as Integer)

Plugin Version: 18.3, 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.

4.48.23 MouseUp(x As Integer, y As Integer)

Plugin Version: 18.3, 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.

4.48.24 Open

Plugin Version: 18.3, 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.

4.48.25 ScaleFactorChanged(NewFactor as double)

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

4.48.26 startRecordingToFileOutput(fileOutput as AVCaptureFileOutputMBS)

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: Informs you that a new media recording should be started.
Notes: If captureFileOutput is an instance of AVCaptureMovieFileOutput this can be achieved by calling startRecordingToOutputFileURL on the captureFileOutput.

4.48.27 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.

4.48.28 Constants

Constants Groups

Constant	Value	Description
ControlsStyleDefault	0	The default controls pane is associated with the view.
ControlsStyleFloating	1	The floating controls pane for media recordings is associated with the view.
ControlsStyleInline	0	The inline controls pane for media recording is associated with the view.
ControlsStyleInlineDeviceSelection	2	The inline controls pane for selection capture devices is associated with th view.

4.49 class AVCompositionMBS

4.49.1 class AVCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVComposition object combines media data from multiple file-based sources in a custom temporal arrangement, in order to present or process media data from multiple sources together.
Notes: All file-based audiovisual assets are eligible to be combined, regardless of container type. The tracks in an AVComposition object are fixed; to change the tracks, you use an instance of its subclass, AVMutableComposition.

At its top-level, AVComposition is a collection of tracks, each presenting media of a specific media type, e.g. audio or video, according to a timeline. Each track is represented by an instance of AVCompositionTrack. Each track is comprised of an array of track segments, represented by instances of AVCompositionTrackSegment. Each segment presents a portion of the media data stored in a source container, specified by URL, a track identifier, and a time mapping. The URL specifies the source container, and the track identifier indicates the track of the source container to be presented.

The time mapping specifies the temporal range of the source track that's to be presented and also specifies the temporal range of its presentation in the composition track. If the durations of the source and destination ranges of the time mapping are the same, the media data for the segment will be presented at its natural rate. Otherwise, the segment will be presented at a rate equal to the ratio source.duration / target.duration.

You can access the track segments of a track using the segments property (an array of AVCompositionTrackSegment objects) of AVCompositionTrack. The collection of tracks with media type information for each, and each with its array of track segments (URL, track identifier, and time mapping), form a complete low-level representation of a composition. This representation can be written out by clients in any convenient form, and subsequently the composition can be reconstituted by instantiating a new AVMutableComposition with AVMutableCompositionTrack objects of the appropriate media type, each with its segments property set according to the stored array of URL, track identifier, and time mapping.

A higher-level interface for constructing compositions is also presented by AVMutableComposition and AVMutableCompositionTrack, offering insertion, removal, and scaling operations without direct manipulation of the trackSegment arrays of composition tracks. This interface makes use of higher-level constructs such as AVAsset and AVAssetTrack, allowing the client to make use of the same references to candidate sources that it would have created in order to inspect or preview them prior to inclusion in a composition.
Subclass of the AVAssetMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.49.2 Methods

4.49.3 CompositionTracks as AVCompositionTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVCompositionTrack objects contained by the composition. (read-only)

4.49.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.49.5 mutableCopy as AVMutableCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an editable copy of the object.

4.49.6 naturalSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the authored size of the visual portion of the composition. (read-only)

4.50 class AVCompositionTrackMBS

4.50.1 class AVCompositionTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The class for a composition track.
Notes: An AVCompositionTrack object provides the low-level representation of tracks a track in an AVComposition object, comprising a media type, a track identifier, and an array of AVCompositionTrackSegment objects, each comprising a URL, and track identifier, and a time mapping.

The timeMapping.target.start of the first track segment in a composition track is kCMTimeZero, and the timeMapping.target.start of each subsequent track segment equals CMTimeRangeGetEnd(<#previousTrackSegment#>.timeMapping.target).

The AVFoundation framework also provides a mutable subclass, AVMutableCompositionTrack.
Subclass of the AVAssetTrackMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.50.2 Methods

4.50.3 CompositionTrackSegments as AVCompositionTrackSegmentMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The composition track's track segments. (read-only)

4.50.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.51 class AVCompositionTrackSegmentMBS

4.51.1 class AVCompositionTrackSegmentMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVCompositionTrackSegment object represents a segment of an AVCompositionTrack object, comprising a URL, and track identifier, and a time mapping from the source track to the composition track.
Notes: You typically use this class to save the low-level representation of a composition to storage formats of your choosing and to reconstitute them from storage.
Subclass of the AVAssetTrackSegmentMBS class.

4.51.2 Methods

4.51.3 compositionTrackSegmentWithTimeRange(timeRange as CMTimeRangeMBS) as AVCompositionTrackSegmentMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a composition track segment that presents an empty track segment.
Notes: timeRange: The time range of the empty composition track segment.

Returns an composition track segment that presents an empty track segment.

This method invokes Constructor with an empty URL, a trackID of kCMPersistentTrackID_Invalid, a time mapping with source.start and source.duration equal to kCMTimeInvalid, and with a target equal to timeRange.

This is the standard low-level representation of an empty track segment.

4.51.4 Constructor(timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a track segment that presents an empty track segment.
Notes: timeRange: The time range of the empty track segment.

This method invokes Constructor with an empty URL, a trackID of kCMPersistentTrackID_Invalid, a time mapping with source.start and source.duration equal to kCMTimeInvalid, and with a target equal to timeRange.

This is the standard low-level representation of an empty track segment.

4.51.5 isEmpty as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the segment is empty. (read-only)
Notes: An empty segment has a valid target time range but sourceURL is nil and the source start time is kCMTimeInvalid; all other fields are undefined.

4.51.6 sourceTrackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The track ID of the container file of the media presented by the track segment. (read-only)

4.51.7 sourceURL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The container file of the media presented by the track segment. (read-only)

4.52 class AVEdgeWidthsMBS

4.52.1 class AVEdgeWidthsMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Defines the thickness of the edge processing region, in pixels.

4.52.2 Properties

4.52.3 Bottom as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The thickness, in pixels, of the bottom-edge processing region of theedgeWidths property.
Notes: (Read only property)

4.52.4 Left as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The thickness, in pixels, of the left-edge processing region of theedgeWidths property.
Notes: (Read only property)

4.52.5 Right as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The thickness, in pixels, of the right-edge processing region of theedgeWidths property.
Notes: (Read only property)

4.52.6 Top as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The thickness, in pixels, of the top-edge processing region of theedgeWidths property.
Notes: (Read only property)

4.53 class AVFoundationMBS

4.53.1 class AVFoundationMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The central AVFoundation class for receiving events and getting constants.
Notes: All events from AVFoundation classes (delegates, notifications) are routed to your subclass of AVFoundationMBS.
Events received on a helper thread a routed to main thread in order to allow you to use Xojo code which is not thread safe.
Please use available property on classes to check if they are available. Some classes are only for Mac OS X 10.8 or future versions.

If you need some new methods/classes in AVFoundation from Mac OS X 10.10 or newer, please email us.
Blog Entries

- MBS Xojo Plugins, version 21.2pr7
- MBS Xojo Plugins, version 20.5pr7
- Write audio file with samples using AVFoundation
- MBS Xojo Plugins, version 19.0pr6
- MBS Xojo Plugins, version 18.3pr6
- MBS Xojo Plugins, version 17.3pr3
- Automated video editing with Xojo
- MBS Xojo / Real Studio Plugins, version 14.3pr8
- MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.0
- MBS Xojo / Real Studio Plugins, version 14.0pr1

Xojo Developer Magazine

- 12.2, page 46: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz
- 12.2, page 44: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz
- 12.2, page 10: News

4.53.2 Methods

4.53.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether AVFoundation is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.53.4 AVAudioBitRateStrategy__Constant as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.5 AVAudioBitRateStrategy__LongTermAverage as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.6 AVAudioBitRateStrategy__Variable as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.7 AVAudioBitRateStrategy__VariableConstrained as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.8 AVAudioFileTypeKey as string

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The audio file type key.
Notes: Value is an integer (audio file type) from AudioFile.h
Available on MacOS 10.13 or newer.

4.53.9 AVAudioTimePitchAlgorithmSpectral as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the time pitch algorithms.
Notes: Highest quality, most computationally expensive. Suitable for music. Variable rate from 1/32 to 32.
Available in OS X v10.9 and later.
Default for Mac OS X.

4.53.10 AVAudioTimePitchAlgorithmTimeDomain as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the time pitch algorithms.
Notes: Modest quality pitch algorithm that is less computationally intensive. Suitable for voice. Variable rate from 1/32 to 32.
Available in OS X v10.9 and later.

4.53.11 AVAudioTimePitchAlgorithmVarispeed as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the values for the time pitch algorithms.
Notes: High quality, no pitch correction. Pitch varies with rate. Variable rate from 1/32 to 32.
Available in OS X v10.9 and later.

4.53.12 AVCaptureDeviceWasConnectedNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Notification that is posted when a new device becomes available.

4.53.13 AVCaptureDeviceWasDisconnectedNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Notification that is posted when an existing device becomes unavailable.

4.53.14 AVCaptureInputPortFormatDescriptionDidChangeNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted if the formatDescription of the capture input port changes.

4.53.15 AVCaptureSessionDidStartRunningNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted when a capture session starts.

4.53.16 AVCaptureSessionDidStopRunningNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted when a capture session stops.

4.53.17 AVCaptureSessionErrorKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Key to retrieve information from a notification from a capture session.
Notes: Key to retrieve the error object from the user info dictionary of an AVCaptureSessionRuntimeErrorNotification.

4.53.18 AVCaptureSessionPreset1280x720 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for 720p quality (1280x720pixel) video output.

4.53.19 AVCaptureSessionPreset320x240 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for 320x240 pixel video output.

4.53.20 AVCaptureSessionPreset352x288 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for CIF quality (352x288 pixel) video output.

4.53.21 AVCaptureSessionPreset640x480 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for VGA quality (640x480 pixel) video output.

4.53.22 AVCaptureSessionPreset960x540 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for quarter HD quality (960x540 pixel) video output.

4.53.23 AVCaptureSessionPresetHigh as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for high quality video and audio output.

4.53.24 AVCaptureSessionPresetiFrame1280x720 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings to achieve 1280x720 quality iFrame H.264 video at about 40 Mbits/sec with AAC audio.
QuickTime movies captured in iFrame format are optimal for editing applications.
Available in OS X v10.9 and later.

4.53.25 AVCaptureSessionPresetiFrame960x540 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings to achieve 960x540 quality iFrame H.264 video at about 30 Mbits/sec with AAC audio.
QuickTime movies captured in iFrame format are optimal for editing applications.
Available in OS X v10.9 and later.

4.53.26 AVCaptureSessionPresetLow as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for output video and audio bitrates suitable for sharing over 3G.

4.53.27 AVCaptureSessionPresetMedium as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for output video and audio bitrates suitable for sharing over WiFi.

4.53.28 AVCaptureSessionPresetPhoto as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to define capture setting presets using the sessionPreset property.
Notes: Specifies capture settings suitable for high resolution photo quality output.

4.53.29 AVCaptureSessionRuntimeErrorNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted if an error occurred during a capture session.
Notes: You retrieve the underlying error from the notification's user info dictionary using the key AVCaptureSessionErrorKey.

4.53.30 AVChannelLayoutKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Key to retrieve channel layout information for playback.
Example:

```
const channels = 2
dim channelLayout as new QTAudioChannelLayoutMBS
channelLayout.ChannelLayoutTag = QTAudioChannelLayoutMBS.kAudioChannelLayoutTag_Stereo

dim outputSettings as new Dictionary

outputSettings.Value(AVFoundationMBS.AVSampleRateKey) = 44100.0
outputSettings.Value(AVFoundationMBS.AVNumberOfChannelsKey) = channels
outputSettings.Value(AVFoundationMBS.AVChannelLayoutKey) = channelLayout.Memory
```

Notes: The corresponding value is an Memoryblock containing an AudioChannelLayout structure.

4.53.31 AVCoreAnimationBeginTimeAtZero as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Support for integration with Core Animation.
Notes: Use this constant to set the CoreAnimation's animation beginTime property to be time 0.
The constant is a small, non-zero, positive value which prevents CoreAnimation from replacing 0.0 with CACurrentMediaTime.

4.53.32 AVEncoderAudioQualityForVBRKey as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the Sample rate converter audio quality settings.

4.53.33 AVEncoderAudioQualityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the audio encoder setting keys for the AVAudioRecorder class.
Notes: Value is a AVAudioQuality* constant.

4.53.34 AVEncoderBitDepthHintKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the audio encoder setting keys for the AVAudioRecorder class.
Notes: An integer ranging from 8 through 32.

4.53.35 AVEncoderBitRateKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the audio encoder setting keys for the AVAudioRecorder class.
Notes: An integer that identifies the audio bit rate.

4.53.36 AVEncoderBitRatePerChannelKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the audio encoder setting keys for the AVAudioRecorder class.
Notes: An integer that identifies the audio bit rate per channel.

4.53.37 AVEncoderBitRateStrategyKey as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the Sample rate converter audio quality settings.

4.53.38 AVErrorDeviceKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is a string with the name of the device.

4.53.39 AVErrorDiscontinuityFlagsKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.

4.53.40 AVErrorFileSizeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is a number containing the size of the file in bytes.

4.53.41 AVErrorMediaSubTypeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is an array of integers that specify media subtypes.
The types are represented by four character codes (4ccs), as defined in CoreAudioTypes.h for audio media and in CMFormatDescription.h for video media.

4.53.42 AVErrorMediaTypeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is a string that specified a media format.

4.53.43 AVErrorPIDKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: nThe corresponding value is a number containing a process ID number.one

4.53.44 AVErrorRecordingSuccessfullyFinishedKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is a Boolean value indicating whether recording finished successfully.

4.53.45 AVErrorTimeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys in the user info dictionary in errors AVFoundation creates.
Notes: The corresponding value is an NSValue object containing a CMTime.

4.53.46 AVFileType3GPP as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.3gpp".
Files are identified with the .3gp, .3gpp, and .sdv extensions.
Available in macOS 10.11 or newer.

4.53.47 AVFileType3GPP2 as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.3gpp2".
Files are identified with the .3g2, .3gp2 extensions.
Available in macOS 10.11 or newer.

4.53.48 AVFileTypeAC3 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the AC-3 audio file format.
The value of this UTI is public.ac3-audio. Files are identified with the .ac3 extension.
Available in OS X v10.9 and later.

4.53.49 AVFileTypeAIFC as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the AIFC audio file format.
The value of this UTI is public.aifc-audio. Files are identified with the .aifc and .cdda extensions.

4.53.50 AVFileTypeAIFF as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the AIFF audio file format.
The value of this UTI is public.aiff-audio. Files are identified with the .aif and .aiff extensions.

4.53.51 AVFileTypeAMR as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the adaptive multi-rate audio file format.
The value of this UTI is org.3gpp.adaptive-multi-rate-audio. Files are identified with the .amr extension.

4.53.52 AVFileTypeAppleM4A as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the Apple m4a audio file format.
The value of this UTI is com.apple.m4a-audio. Files are identified with the .m4a extension.

4.53.53 AVFileTypeAppleM4V as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the iTunes video file format.
The value of this UTI is com.apple.mpeg-4-video. Files are identified with the .m4v extension.

4.53.54 AVFileTypeAVCI as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.avci".
Files are identified with the .avci extension.
Available in macOS 10.13 or newer.

4.53.55 AVFileTypeCoreAudioFormat as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the CoreAudio file format.
The value of this UTI is com.apple.coreaudio-format. Files are identified with the .caf extension.

4.53.56 AVFileTypeDNG as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "com.adobe.raw-image".
Files are identified with the .dng extension.
Available in macOS 10.13 or newer.

4.53.57 AVFileTypeEnhancedAC3 as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.ac3-audio".
Files are identified with the .ac3 extension.
Available in macOS 10.9 or newer.

4.53.58 AVFileTypeHEIC as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.heic".
Files are identified with the .heic extension.

Available in macOS 10.13 or newer.

4.53.59 AVFileTypeHEIF as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.heif".
Files are identified with the .heif extension.
Available in macOS 10.13 or newer.

4.53.60 AVFileTypeJPEG as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.jpeg".
Files are identified with the .jpg or .jpeg extension.
Available in macOS 10.13 or newer.

4.53.61 AVFileTypeMPEG4 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the MPEG-4 file format.
The value of this UTI is public.mpeg-4. Files are identified with the .mp4 extension.

4.53.62 AVFileTypeMPEGLayer3 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the MPEG layer 3 audio file format.
The value of this UTI is public.mp3. Files are identified with the .mp3 extension.
Available in OS X v10.9 and later.

4.53.63 AVFileTypeQuickTimeMovie as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the QuickTime movie file format.
The value of this UTI is com.apple.quicktime-movie. Files are identified with the .mov and .qt extensions.

4.53.64 AVFileTypeSunAU as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: UTI for the Sun/NeXT audio file format.
The value of this UTI is public.au-audio. Files are identified with the .au and .snd extensions.
Available in OS X v10.9 and later.

4.53.65 AVFileTypeTIFF as string

Plugin Version: 17.3, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: The value of this UTI is "public.tiff".
Files are identified with the .tiff or .tif extension.
Available in macOS 10.13 or newer.

4.53.66 AVFileTypeWAVE as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the file format uniform type identifiers (UTIs).
Notes: A UTI for the WAVE audio file format.
The value of this UTI is com.microsoft.waveform-audio. Files are identified with the .wav, .wave, and .bwf extensions.

4.53.67 AVFormatIDKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to all audio formats handled by the AVAudioPlayer and AVAudioRecorder classes.
Notes: A format identifier. See the "Audio Data Format Identifiers" enumeration in Core Audio Data

Types Reference.

4.53.68 AVFoundationErrorDomain as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Domain for AVFoundation errors.

4.53.69 AVLayerVideoGravityResize as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constants define how the video is displayed within a layer's bounds rectangle.
Notes: Specifies that the video should be stretched to fill the layer's bounds.

4.53.70 AVLayerVideoGravityResizeAspect as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constants define how the video is displayed within a layer's bounds rectangle.
Notes: Specifies that the player should preserve the video's aspect ratio and fit the video within the layer's bounds.

4.53.71 AVLayerVideoGravityResizeAspectFill as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constants define how the video is displayed within a layer's bounds rectangle.
Notes: Specifies that the player should preserve the video's aspect ratio and fill the layer's bounds.

4.53.72 AVLinearPCMBitDepthKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to linear PCM audio formats.
Notes: An integer that indicates the bit depth for a linear PCM audio format—one of 8, 16, 24, or 32.

4.53.73 AVLinearPCMIsBigEndianKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to linear PCM audio formats.
Notes: A Boolean value that indicates whether the audio format is big endian (true) or little endian (false).

4.53.74 AVLinearPCMIsFloatKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to linear PCM audio formats.
Notes: A Boolean value that indicates that the audio format is floating point (true) or fixed point (false).

4.53.75 AVLinearPCMIsNonInterleaved as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to linear PCM audio formats.
Notes: A Boolean value that indicates that the audio format is non-interleaved (true) or interleaved (false).

4.53.76 AVMakeRectWithAspectRatioInsideRect(aspectRatio as CGSizeMBS, boundingRect as CGRectMBS) as CGRectMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a scaled CGRect that maintains the aspect ratio specified by a CGSize within a bounding CGRect.
Notes: aspectRatio: The width and height ratio (aspect ratio) you want to maintain.
boundingRect: The bounding rectangle you want to fit into.

Returns a scaled CGRect that maintains the aspect ratio specified by aspectRatio that fits withinboundingRect.

This is useful when attempting to fit the naturalSize property of an AVPlayerItem object within the bounds of another CALayer. You would typically use the return value of this function as an AVPlayerLayer frame property value. For example:

myPlayerLayer.frame = AVMakeRectWithAspectRatioInsideRect(myPlayerItem.naturalSize, mySuperLayer.bounds)

4.53.77 AVMediaCharacteristicAudible as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the characteristics of media types.
Notes: Indicates that the media is audible.

4.53.78 AVMediaCharacteristicContainsAlphaChannel as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: To determine whether alpha is straight or pre-multiplied, look for the format description extension with key kCMFormatDescriptionExtension_AlphaChannelMode.

4.53.79 AVMediaCharacteristicContainsOnlyForcedSubtitles as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the options presents only forced subtitles.
Media options with forced-only subtitles are typically selected when 1) the user has not selected a legible option with an accessibility characteristic or an auxiliary purpose and 2) its locale matches the locale of the selected audible media selection option.
The value of this characteristic is ”public.subtitles.forced-only”.
The presence of this characteristic for a legible media option is inferred from the format description of the associated track that presents the subtitle media.
Available in OS X v10.8 and later.

4.53.80 AVMediaCharacteristicDescribesMusicAndSoundForAccessibility as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option includes legible content in the language of its specified locale that describes music and sound effects occurring in program audio.
It is possible for a legible media option to include both transcriptions of spoken dialog and descriptions of music and sound effects.
The value of this characteristic is ”public.accessibility.describes-music-and-sound”.
For QuickTime movie and .m4v files, a media option is considered to have the characteristic AVMediaCharacteristicDescribesMusicAndSoundForAccessibility only if it’s explicitly tagged with that characteristic.
Available in OS X v10.8 and later.

4.53.81 AVMediaCharacteristicDescribesVideoForAccessibility as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option includes audible content that describes the visual portion of the presentation.
It is possible for a legible media option to include both transcriptions of spoken dialog and descriptions of music and sound effects.
The value of this characteristic is "public.accessibility.describes-video".
For QuickTime movie and .m4v files a media option is considered to have the characteristic AVMediaCharacteristicDescribesVideoForAccessibility only if it's explicitly tagged with that characteristic.
Available in OS X v10.8 and later.

4.53.82 AVMediaCharacteristicDubbedTranslation as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: The track or media selection option contains a language or dialect translation of originally or previously produced content.

4.53.83 AVMediaCharacteristicEasyToRead as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option provides legible content in the language of its specified locale and that the content has been edited for ease of reading.
Closed caption tracks that carry "easy reader" captions (per the CEA-608 specification) should be tagged with this characteristic. Subtitle tracks can also be tagged with this characteristic, where appropriate.
The value of this characteristic is "public.easy-to-read".
For QuickTime movie and .m4v files a media option is considered to have the characteristic AVMediaCharacteristicEasyToRead only if it's explicitly tagged with that characteristic.

4.53.84 AVMediaCharacteristicFrameBased as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the characteristics of media types.
Notes: Indicates that the media is frame-based.

4.53.85 AVMediaCharacteristicIsAuxiliaryContent as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option includes content that's marked by the content author as auxiliary to the presentation of the asset.
Example: an option that presents audio media containing commentary on the presentation would typically have this characteristic.
The value of this characteristic is "public.auxiliary-content".
For QuickTime movie and .m4v files, a media option is considered to have the characteristic AVMediaCharacteristicIsAuxiliaryContent if it's explicitly tagged with that characteristic or if, as a member of an alternate track group, its associated track is excluded from autoselection.
Available in OS X v10.8 and later.

4.53.86 AVMediaCharacteristicIsMainProgramContent as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option includes content that's marked by the content author as intrinsic to the presentation of the asset.
Example: an option that presents the main program audio for the presentation, regardless of locale, would typically have this characteristic.
The value of this characteristic is "public.main-program-content".
The presence of this characteristic for a media option is inferred; any option that does not have the characteristic AVMediaCharacteristicIsAuxiliaryContent is considered to have the characteristic.
Available in OS X v10.8 and later.

4.53.87 AVMediaCharacteristicIsOriginalContent as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: A media characteristic that indicates that a track or media selection option includes content that's marked by the content author as original to the principal production of the media, as opposed to supplementary or derivative content created by means of language translation or by other means.

The value of this characteristic is @"public.original-content".

Example: an option that presents audio media with dialog in the principal language of the production may be tagged with this characteristic; audio media containing dialog dubbed in a language other than the principal language of the production typically would not be tagged with this characteristic.

Note for content authors: for QuickTime movie and .m4v files and for HTTP Live Streaming, a media option is considered to have the characteristic AVMediaCharacteristicIsOriginalContent only if it's explicitly tagged with the characteristic.

See the discussion of the tagging of tracks with media characteristics below.

4.53.88 AVMediaCharacteristicLanguageTranslation as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: A media characteristic that indicates that a track or media selection option contains a language or dialect translation of originally or previously produced content, intended to be used as a substitute for that content by users who prefer its designated language.

The value of this characteristic is "public.translation".

Note for content authors: for QuickTime movie and .m4v files a media option is considered to have the characteristic AVMediaCharacteristicLanguageTranslation only if it's explicitly tagged with that characteristic.

4.53.89 AVMediaCharacteristicLegible as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the characteristics of media types.
Notes: Indicates that the media is legible.

4.53.90 AVMediaCharacteristicTranscribesSpokenDialogForAccessibility as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: Indicates that the option includes legible content in the language of its specified locale that transcribes spoken dialog.
It is possible for a legible media option to include both transcriptions of spoken dialog and descriptions of music and sound effects.
The value of this characteristic is "public.accessibility.transcribes-spoken-dialog".
For QuickTime movie and .m4v files, a media option is considered to have the characteristic AVMediaCharacteristicTranscribesSpokenDialogForAccessibility only if it's explicitly tagged with that characteristic.
Available in OS X v10.8 and later.

4.53.91 AVMediaCharacteristicUsesWideGamutColorSpace as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: A media characteristic that indicates that a track uses a wide gamut color space and therefore may make use of colors that cannot be accurately represented otherwise.

A wide color space such as AVVideo*_P3_D65 contains additional dynamic range that may benefit from special treatment when compositing. Care should be taken to avoid clamping. Non-wide spaces include AVVideo*_ITU_R_709_2 and AVVideo*_SMPTE_C.

4.53.92 AVMediaCharacteristicVisual as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the characteristics of media types.
Notes: Indicates that the media is visual.

4.53.93 AVMediaCharacteristicVoiceOverTranslation as string

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

Function: One of the media characteristic that may be present in an AVMediaSelectionOption object.
Notes: A media characteristic that indicates that a track or media selection option contains a language translation of originally or previously produced content, created by adding, in its designated language, a verbal interpretation of dialog and translations of other important information to a new mix of the audio content.

Tracks to which this characteristic is assigned should typically also be assigned the characteristic AVMediaCharacteristicLanguageTranslation.

The value of this characteristic is ”public.translation.voice-over”.

Note for content authors: for QuickTime movie and .m4v files a media option is considered to have the characteristic AVMediaCharacteristicVoiceOverTranslation only if it’s explicitly tagged with that characteristic.

4.53.94 AVMediaTypeAudio as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.

Notes: Specifies audio.

4.53.95 AVMediaTypeClosedCaption as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies closed-caption content.

4.53.96 AVMediaTypeDepthData as string

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

Function: The key for media type depth data.

4.53.97 AVMediaTypeMetadata as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies metadata.

4.53.98 AVMediaTypeMetadataObject as string

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

Function: mediaType of AVCaptureInputPorts that provide AVMetadataObjects.
Notes: Prior to iOS 9.0, camera AVCaptureDeviceInputs provide metadata (detected faces and barcodes) to an AVCaptureMetadataOutput through an AVCaptureInputPort whose mediaType is AVMediaTypeMetadata. The AVCaptureMetadataOutput presents metadata to the client as an array of AVMetadataObjects, which are defined by Apple and not externally subclassable. Starting in iOS 9.0, clients may record arbitrary metadata to a movie file using the AVCaptureMovieFileOutput. The movie file output consumes metadata in a different format than the AVCaptureMetadataOutput, namely it accepts CMSampleBuffers of type 'meta'.

4.53.99 AVMediaTypeMuxed as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies muxed media.

4.53.100 AVMediaTypeSubtitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies subtitles.

4.53.101 AVMediaTypeText as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies text.

4.53.102 AVMediaTypeTimecode as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies a time code.

4.53.103 AVMediaTypeVideo as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the media types.
Notes: Specifies video.

4.53.104 AVMetadata3GPUserDataKeyAlbumAndTrack as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.105 AVMetadata3GPUserDataKeyAuthor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.106 AVMetadata3GPUserDataKeyCollection as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.107 AVMetadata3GPUserDataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.108 AVMetadata3GPUserDataKeyDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.109 AVMetadata3GPUserDataKeyGenre as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.110 AVMetadata3GPUserDataKeyKeywordList as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.111 AVMetadata3GPUserDataKeyLocation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.112 AVMetadata3GPUserDataKeyMediaClassification as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.113 AVMetadata3GPUserDataKeyMediaRating as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.114 AVMetadata3GPUserDataKeyPerformer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.115 AVMetadata3GPUserDataKeyRecordingYear as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.116 AVMetadata3GPUserDataKeyThumbnail as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.117 AVMetadata3GPUserDataKeyTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.118 AVMetadata3GPUserDataKeyUserRating as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.119 AVMetadataCommonKeyAlbumName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.120 AVMetadataCommonKeyArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.121 AVMetadataCommonKeyArtwork as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.122 AVMetadataCommonKeyAuthor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.123 AVMetadataCommonKeyContributor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.124 AVMetadataCommonKeyCopyrights as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.125 AVMetadataCommonKeyCreationDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.126 AVMetadataCommonKeyCreator as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.127 AVMetadataCommonKeyDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.128 AVMetadataCommonKeyFormat as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.129 AVMetadataCommonKeyIdentifier as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.130 AVMetadataCommonKeyLanguage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.131 AVMetadataCommonKeyLastModifiedDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.132 AVMetadataCommonKeyLocation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.133 AVMetadataCommonKeyMake as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.134 AVMetadataCommonKeyModel as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.135 AVMetadataCommonKeyPublisher as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.136 AVMetadataCommonKeyRelation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.137 AVMetadataCommonKeySoftware as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.138 AVMetadataCommonKeySource as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.139 AVMetadataCommonKeySubject as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.140 AVMetadataCommonKeyTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.141 AVMetadataCommonKeyType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the common metadata keys.

4.53.142 AVMetadataFormatID3Metadata as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the metadata formats.
Notes: The asset has metadata in the ID3 format.

4.53.143 AVMetadataFormatISOUserData as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the metadata formats.

4.53.144 AVMetadataFormatiTunesMetadata as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the metadata formats.
Notes: The asset has metadata in the iTunes format.

4.53.145 AVMetadataFormatQuickTimeMetadata as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the metadata formats.
Notes: The asset has metadata in the QuickTime metadata format.

4.53.146 AVMetadataFormatQuickTimeUserData as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the metadata formats.
Notes: The asset has metadata in the QuickTime user data format.

4.53.147 AVMetadataID3MetadataKeyAlbumSortOrder as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSOA album sort order.

4.53.148 AVMetadataID3MetadataKeyAlbumTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TALB album/Movie/Show title.

4.53.149 AVMetadataID3MetadataKeyAttachedPicture as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: APIC attached picture.

4.53.150 AVMetadataID3MetadataKeyAudioEncryption as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: AENC audio encryption.

4.53.151 AVMetadataID3MetadataKeyAudioSeekPointIndex as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: ASPI audio seek point index.

4.53.152 AVMetadataID3MetadataKeyBand as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPE2 band/orchestra/accompaniment.

4.53.153 AVMetadataID3MetadataKeyBeatsPerMinute as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TBPM BPM (beats per minute).

4.53.154 AVMetadataID3MetadataKeyComments as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: COMM comments.

4.53.155 AVMetadataID3MetadataKeyCommercial as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: COMR commercial frame.

4.53.156 AVMetadataID3MetadataKeyCommercialInformation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WCOM commercial information.

4.53.157 AVMetadataID3MetadataKeyComposer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TCOM composer.

4.53.158 AVMetadataID3MetadataKeyConductor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPE3 conductor/performer refinement.

4.53.159 AVMetadataID3MetadataKeyContentGroupDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TIT1 content group description.

4.53.160 AVMetadataID3MetadataKeyContentType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TCON content type.

4.53.161 AVMetadataID3MetadataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TCOP copyright message.

4.53.162 AVMetadataID3MetadataKeyCopyrightInformation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WCOP copyright/legal information.

4.53.163 AVMetadataID3MetadataKeyDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDAT date.

4.53.164 AVMetadataID3MetadataKeyEncodedBy as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TENC encoded by.

4.53.165 AVMetadataID3MetadataKeyEncodedWith as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSSE software/hardware and settings used for encoding.

4.53.166 AVMetadataID3MetadataKeyEncodingTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDEN encoding time.

4.53.167 AVMetadataID3MetadataKeyEncryption as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: ENCR encryption method registration.

4.53.168 AVMetadataID3MetadataKeyEqualization as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: EQUA equalization.

4.53.169 AVMetadataID3MetadataKeyEqualization2 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: EQU2 equalization (2).

4.53.170 AVMetadataID3MetadataKeyEventTimingCodes as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: ETCO event timing codes.

4.53.171 AVMetadataID3MetadataKeyFileOwner as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TOWN file owner/licensee.

4.53.172 AVMetadataID3MetadataKeyFileType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TFLT file type.

4.53.173 AVMetadataID3MetadataKeyGeneralEncapsulatedObject as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: GEOB general encapsulated object.

4.53.174 AVMetadataID3MetadataKeyGroupIdentifier as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: GRID group identification registration.

4.53.175 AVMetadataID3MetadataKeyInitialKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TKEY initial key.

4.53.176 AVMetadataID3MetadataKeyInternationalStandardRecordingCode as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSRC ISRC (international standard recording code).

4.53.177 AVMetadataID3MetadataKeyInternetRadioStationName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TRSN internet radio station name.

4.53.178 AVMetadataID3MetadataKeyInternetRadioStationOwner as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TRSO internet radio station owner.

4.53.179 AVMetadataID3MetadataKeyInvolvedPeopleList_v23 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: IPLS involved people list.

4.53.180 AVMetadataID3MetadataKeyInvolvedPeopleList_v24 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TIPL involved people list.

4.53.181 AVMetadataID3MetadataKeyLanguage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TLAN language(s).

4.53.182 AVMetadataID3MetadataKeyLeadPerformer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPE1 lead performer(s)/Soloist(s).

4.53.183 AVMetadataID3MetadataKeyLength as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TLEN length.

4.53.184 AVMetadataID3MetadataKeyLink as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: LINK linked information.

4.53.185 AVMetadataID3MetadataKeyLyricist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TEXT lyricist/text writer.

4.53.186 AVMetadataID3MetadataKeyMediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TMED media type.

4.53.187 AVMetadataID3MetadataKeyModifiedBy as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPE4 interpreted, remixed, or otherwise modified by.

4.53.188 AVMetadataID3MetadataKeyMood as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TMOO mood.

4.53.189 AVMetadataID3MetadataKeyMPEGLocationLookupTable as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: MLLT MPEG location lookup table.

4.53.190 AVMetadataID3MetadataKeyMusicCDIdentifier as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: MCDI music CD identifier.

4.53.191 AVMetadataID3MetadataKeyMusicianCreditsList as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TMCL musician credits list.

4.53.192 AVMetadataID3MetadataKeyOfficialArtistWebpage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WOAR official artist/performer webpage.

4.53.193 AVMetadataID3MetadataKeyOfficialAudioFileWebpage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WOAF official audio file webpage.

4.53.194 AVMetadataID3MetadataKeyOfficialAudioSourceWebpage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WOAS official audio source webpage.

4.53.195 AVMetadataID3MetadataKeyOfficialInternetRadioStationHomepage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WORS official Internet radio station homepage.

4.53.196 AVMetadataID3MetadataKeyOfficialPublisherWebpage as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WPUB publishers official webpage.

4.53.197 AVMetadataID3MetadataKeyOriginalAlbumTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TOAL original album/movie/show title.

4.53.198 AVMetadataID3MetadataKeyOriginalArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TOPE original artist(s)/performer(s).

4.53.199 AVMetadataID3MetadataKeyOriginalFilename as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TOFN original filename.

4.53.200 AVMetadataID3MetadataKeyOriginalLyricist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TOLY original lyricist(s)/text writer(s).

4.53.201 AVMetadataID3MetadataKeyOriginalReleaseTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDOR original release time.

4.53.202 AVMetadataID3MetadataKeyOriginalReleaseYear as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TORY original release year.

4.53.203 AVMetadataID3MetadataKeyOwnership as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: OWNE ownership frame.

4.53.204 AVMetadataID3MetadataKeyPartOfASet as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPOS part of a set.

4.53.205 AVMetadataID3MetadataKeyPayment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WPAY payment.

4.53.206 AVMetadataID3MetadataKeyPerformerSortOrder as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSOP performer sort order.

4.53.207 AVMetadataID3MetadataKeyPlayCounter as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: PCNT play counter.

4.53.208 AVMetadataID3MetadataKeyPlaylistDelay as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDLY playlist delay.

4.53.209 AVMetadataID3MetadataKeyPopularimeter as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: POPM popularimeter.

4.53.210 AVMetadataID3MetadataKeyPositionSynchronization as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: POSS position synchronisation frame.

4.53.211 AVMetadataID3MetadataKeyPrivate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: PRIV private frame.

4.53.212 AVMetadataID3MetadataKeyProducedNotice as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPRO produced notice.

4.53.213 AVMetadataID3MetadataKeyPublisher as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TPUB publisher.

4.53.214 AVMetadataID3MetadataKeyRecommendedBufferSize as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: RBUF recommended buffer size.

4.53.215 AVMetadataID3MetadataKeyRecordingDates as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TRDA recording dates.

4.53.216 AVMetadataID3MetadataKeyRecordingTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDRC recording time.

4.53.217 AVMetadataID3MetadataKeyRelativeVolumeAdjustment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: RVAD relative volume adjustment.

4.53.218 AVMetadataID3MetadataKeyRelativeVolumeAdjustment2 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: RVA2 relative volume adjustment (2).

4.53.219 AVMetadataID3MetadataKeyReleaseTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDRL release time.

4.53.220 AVMetadataID3MetadataKeyReverb as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: RVRB reverb.

4.53.221 AVMetadataID3MetadataKeySeek as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: SEEK seek frame.

4.53.222 AVMetadataID3MetadataKeySetSubtitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSST set subtitle.

4.53.223 AVMetadataID3MetadataKeySignature as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: SIGN signature frame.

4.53.224 AVMetadataID3MetadataKeySize as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSIZ size.

4.53.225 AVMetadataID3MetadataKeySubTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TIT3 subtitle/description refinement.

4.53.226 AVMetadataID3MetadataKeySynchronizedLyric as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: SYLT synchronized lyric/text.

4.53.227 AVMetadataID3MetadataKeySynchronizedTempoCodes as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: SYTC synchronized tempo codes.

4.53.228 AVMetadataID3MetadataKeyTaggingTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TDTG tagging time.

4.53.229 AVMetadataID3MetadataKeyTermsOfUse as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: USER terms of use.

4.53.230 AVMetadataID3MetadataKeyTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TIME time.

4.53.231 AVMetadataID3MetadataKeyTitleDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TIT2 title/songname/content description.

4.53.232 AVMetadataID3MetadataKeyTitleSortOrder as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TSOT title sort order.

4.53.233 AVMetadataID3MetadataKeyTrackNumber as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TRCK track number/position in set.

4.53.234 AVMetadataID3MetadataKeyUniqueFileIdentifier as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: UFID unique file identifier.

4.53.235 AVMetadataID3MetadataKeyUnsynchronizedLyric as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: USLT unsynchronized lyric/text transcription.

4.53.236 AVMetadataID3MetadataKeyUserText as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TXXX user defined text information frame.

4.53.237 AVMetadataID3MetadataKeyUserURL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: WXXX user defined URL link frame.

4.53.238 AVMetadataID3MetadataKeyYear as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for ID3 Metadata.
Notes: TYER year.

4.53.239 AVMetadataISOUserDataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.240 AVMetadataiTunesMetadataKeyAccountKind as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.241 AVMetadataiTunesMetadataKeyAcknowledgement as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.242 AVMetadataiTunesMetadataKeyAlbum as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.243 AVMetadataiTunesMetadataKeyAlbumArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.244 AVMetadataiTunesMetadataKeyAppleID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.245 AVMetadataiTunesMetadataKeyArranger as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.246 AVMetadataiTunesMetadataKeyArtDirector as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.247 AVMetadataiTunesMetadataKeyArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.248 AVMetadataiTunesMetadataKeyArtistID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.249 AVMetadataiTunesMetadataKeyAuthor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.250 AVMetadataiTunesMetadataKeyBeatsPerMin as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.251 AVMetadataiTunesMetadataKeyComposer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.252 AVMetadataiTunesMetadataKeyConductor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.253 AVMetadataiTunesMetadataKeyContentRating as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.254 AVMetadataiTunesMetadataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.255 AVMetadataiTunesMetadataKeyCoverArt as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.
Example:

```
// load file and look for cover art

Dim file As FolderItem = SpecialFolder.Desktop.Child("test.m4a")
Dim a As AVURLAssetMBS = AVURLAssetMBS.URLAssetWithFile(file)

Dim iTunesMetadataKeyCoverArtString As String = AVFoundationMBS.AVMetadataiTunesMetadataKeyCoverArt
Dim iTunesMetadataKeyCoverArtNumber As Integer = OSTypeFromStringMBS(iTunesMetadataKeyCoverArtString)

Dim items() As AVMetadataItemMBS = a.metadata

For Each item As AVMetadataItemMBS In items

Dim key As Variant = item.key
Dim Val As Variant = item.value

If key.IntegerValue = iTunesMetadataKeyCoverArtNumber Or key.StringValue = iTunesMetadataKeyCoverArtString Then

Dim v As MemoryBlock = Val
window1.Backdrop = picture.FromData(v)
End If

Next
```

4.53.256 AVMetadataiTunesMetadataKeyCredits as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.257 AVMetadataiTunesMetadataKeyDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.258 AVMetadataiTunesMetadataKeyDirector as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.259 AVMetadataiTunesMetadataKeyDiscCompilation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.260 AVMetadataiTunesMetadataKeyDiscNumber as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.261 AVMetadataiTunesMetadataKeyEncodedBy as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.262 AVMetadataiTunesMetadataKeyEncodingTool as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.263 AVMetadataiTunesMetadataKeyEQ as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.264 AVMetadataiTunesMetadataKeyExecProducer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.265 AVMetadataiTunesMetadataKeyGenreID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.266 AVMetadataiTunesMetadataKeyGrouping as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.267 AVMetadataiTunesMetadataKeyLinerNotes as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.268 AVMetadataiTunesMetadataKeyLyrics as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.269 AVMetadataiTunesMetadataKeyOnlineExtras as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.270 AVMetadataiTunesMetadataKeyOriginalArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.271 AVMetadataiTunesMetadataKeyPerformer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.272 AVMetadataiTunesMetadataKeyPhonogramRights as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.273 AVMetadataiTunesMetadataKeyPlaylistID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.274 AVMetadataiTunesMetadataKeyPredefinedGenre as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.275 AVMetadataiTunesMetadataKeyProducer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.276 AVMetadataiTunesMetadataKeyPublisher as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.277 AVMetadataiTunesMetadataKeyRecordCompany as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.278 AVMetadataiTunesMetadataKeyReleaseDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.279 AVMetadataiTunesMetadataKeySoloist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.280 AVMetadataiTunesMetadataKeySongID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.281 AVMetadataiTunesMetadataKeySongName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.282 AVMetadataiTunesMetadataKeySoundEngineer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.283 AVMetadataiTunesMetadataKeyThanks as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.284 AVMetadataiTunesMetadataKeyTrackNumber as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.285 AVMetadataiTunesMetadataKeyTrackSubTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.286 AVMetadataiTunesMetadataKeyUserComment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.287 AVMetadataiTunesMetadataKeyUserGenre as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for iTunes Metadata.

4.53.288 AVMetadataKeySpaceCommon as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the key spaces.
Notes: The common key space. Keys in this space represent standard versions of keys that are found in most or all other key spaces.

4.53.289 AVMetadataKeySpaceID3 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the key spaces.
Notes: The ID3 key space.

4.53.290 AVMetadataKeySpaceISOUserData as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the metadata formats.

4.53.291 AVMetadataKeySpaceiTunes as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the key spaces.
Notes: The iTunes key space.

4.53.292 AVMetadataKeySpaceQuickTimeMetadata as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the key spaces.
Notes: The QuickTime metadata key space.

4.53.293 AVMetadataKeySpaceQuickTimeUserData as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the key spaces.
Notes: The QuickTime user data key space.

4.53.294 AVMetadataQuickTimeMetadataKeyAlbum as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.295 AVMetadataQuickTimeMetadataKeyArranger as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.296 AVMetadataQuickTimeMetadataKeyArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.297 AVMetadataQuickTimeMetadataKeyArtwork as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.298 AVMetadataQuickTimeMetadataKeyAuthor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.299 AVMetadataQuickTimeMetadataKeyCameraFrameReadoutTime as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.300 AVMetadataQuickTimeMetadataKeyCameraIdentifier as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.301 AVMetadataQuickTimeMetadataKeyCollectionUser as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.302 AVMetadataQuickTimeMetadataKeyComment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.303 AVMetadataQuickTimeMetadataKeyComposer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.304 AVMetadataQuickTimeMetadataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.305 AVMetadataQuickTimeMetadataKeyCreationDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.306 AVMetadataQuickTimeMetadataKeyCredits as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.307 AVMetadataQuickTimeMetadataKeyDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.308 AVMetadataQuickTimeMetadataKeyDirectionFacing as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.309 AVMetadataQuickTimeMetadataKeyDirectionMotion as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.310 AVMetadataQuickTimeMetadataKeyDirector as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.311 AVMetadataQuickTimeMetadataKeyDisplayName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.312 AVMetadataQuickTimeMetadataKeyEncodedBy as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.313 AVMetadataQuickTimeMetadataKeyGenre as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.314 AVMetadataQuickTimeMetadataKeyInformation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.315 AVMetadataQuickTimeMetadataKeyiXML as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.316 AVMetadataQuickTimeMetadataKeyKeywords as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.317 AVMetadataQuickTimeMetadataKeyLocationBody as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.318 AVMetadataQuickTimeMetadataKeyLocationDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.319 AVMetadataQuickTimeMetadataKeyLocationISO6709 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.320 AVMetadataQuickTimeMetadataKeyLocationName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.321 AVMetadataQuickTimeMetadataKeyLocationNote as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.322 AVMetadataQuickTimeMetadataKeyLocationRole as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.323 AVMetadataQuickTimeMetadataKeyMake as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.324 AVMetadataQuickTimeMetadataKeyModel as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.325 AVMetadataQuickTimeMetadataKeyOriginalArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.326 AVMetadataQuickTimeMetadataKeyPerformer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.327 AVMetadataQuickTimeMetadataKeyPhonogramRights as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.328 AVMetadataQuickTimeMetadataKeyProducer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.329 AVMetadataQuickTimeMetadataKeyPublisher as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.330 AVMetadataQuickTimeMetadataKeyRatingUser as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.331 AVMetadataQuickTimeMetadataKeySoftware as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.332 AVMetadataQuickTimeMetadataKeyTitle as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.333 AVMetadataQuickTimeMetadataKeyYear as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible keys for QuickTime Metadata.

4.53.334 AVMetadataQuickTimeUserDataKeyAlbum as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.335 AVMetadataQuickTimeUserDataKeyArranger as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.336 AVMetadataQuickTimeUserDataKeyArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.337 AVMetadataQuickTimeUserDataKeyAuthor as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.338 AVMetadataQuickTimeUserDataKeyChapter as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.339 AVMetadataQuickTimeUserDataKeyComment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.340 AVMetadataQuickTimeUserDataKeyComposer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.341 AVMetadataQuickTimeUserDataKeyCopyright as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.342 AVMetadataQuickTimeUserDataKeyCreationDate as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.343 AVMetadataQuickTimeUserDataKeyCredits as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.344 AVMetadataQuickTimeUserDataKeyDescription as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.345 AVMetadataQuickTimeUserDataKeyDirector as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.346 AVMetadataQuickTimeUserDataKeyDisclaimer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.347 AVMetadataQuickTimeUserDataKeyEncodedBy as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.348 AVMetadataQuickTimeUserDataKeyFullName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.349 AVMetadataQuickTimeUserDataKeyGenre as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.350 AVMetadataQuickTimeUserDataKeyHostComputer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.351 AVMetadataQuickTimeUserDataKeyInformation as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.352 AVMetadataQuickTimeUserDataKeyKeywords as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.353 AVMetadataQuickTimeUserDataKeyLocationISO6709 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.354 AVMetadataQuickTimeUserDataKeyMake as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.355 AVMetadataQuickTimeUserDataKeyModel as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.356 AVMetadataQuickTimeUserDataKeyOriginalArtist as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.357 AVMetadataQuickTimeUserDataKeyOriginalFormat as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.358 AVMetadataQuickTimeUserDataKeyOriginalSource as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.359 AVMetadataQuickTimeUserDataKeyPerformers as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.360 AVMetadataQuickTimeUserDataKeyPhonogramRights as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.361 AVMetadataQuickTimeUserDataKeyProducer as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.362 AVMetadataQuickTimeUserDataKeyProduct as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.363 AVMetadataQuickTimeUserDataKeyPublisher as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.364 AVMetadataQuickTimeUserDataKeySoftware as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.365 AVMetadataQuickTimeUserDataKeySpecialPlaybackRequirements as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.366 AVMetadataQuickTimeUserDataKeyTaggedCharacteristic as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.367 AVMetadataQuickTimeUserDataKeyTrack as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.368 AVMetadataQuickTimeUserDataKeyTrackName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.369 AVMetadataQuickTimeUserDataKeyURLLink as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.370 AVMetadataQuickTimeUserDataKeyWarning as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.371 AVMetadataQuickTimeUserDataKeyWriter as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the possible metadata keys for QuickTime UserData.

4.53.372 AVNumberOfChannelsKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to all audio formats handled by the AVAudioPlayer and AVAudioRecorder classes.
Notes: The number of channels expressed as an integer value.

4.53.373 AVPlayerItemDidPlayToEndTimeNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted when the item has played to its end time.
Notes: The notification's object is the item that finished playing.

4.53.374 AVPlayerItemFailedToPlayToEndTimeErrorKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Key to retrieve information from a notification's user info dictionary.
Notes: The key to retrieve an error object (NSError) from the user info dictionary of an AVPlayerItem-

TimeJumpedNotification notification.

4.53.375 AVPlayerItemFailedToPlayToEndTimeNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted when the item failed to play to its end time.
Notes: The notification's object is the item that finished playing.
The user info dictionary contains an error object that describes the problem—seeAVPlayerItemFailedToPlayToEndTimeErrorKey.

4.53.376 AVPlayerItemNewAccessLogEntryNotification as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The notification name to use with NSNotificationObserverMBS class.
Notes: Posted when a new access log entry has been added.
The notification's object is the player item. The new log entry is accessible via accessLog.
Available on Mac OS X 10.9 or later.

4.53.377 AVPlayerItemNewErrorLogEntryNotification as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The notification name to use with NSNotificationObserverMBS class.
Notes: Posted when a new error log entry has been added.
The notification's object is the player item. The new log entry is accessible via errorLog, respectively.
Available on Mac OS X 10.9 or later.

4.53.378 AVPlayerItemPlaybackStalledNotification as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The notification name to use with NSNotificationObserverMBS class.
Notes: Posted when some media did not arrive in time to continue playback.
The notification's object is the AVPlayerItem instance whose playback was unable to continue because the necessary media isn't available yet, usually because it didn't arrive in a timely fashion from its source over a network. Playback will continue once a sufficient amount of media has subsequently been delivered.
Available on Mac OS X 10.9 or later.

4.53.379 AVPlayerItemTimeJumpedNotification as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Posted when the item's current time has changed discontinuously.
Notes: The notification's object is the item.

4.53.380 AVSampleRateConverterAlgorithmKey as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The algorithm for the rate conversion.
Notes: Can be AVSampleRateConverterAlgorithm_Mastering or AVSampleRateConverterAlgorithm_Normal.
Available in OS X v10.9 and later.

4.53.381 AVSampleRateConverterAlgorithm_Mastering as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the supported values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.382 AVSampleRateConverterAlgorithm_Normal as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the supported values for the AVEncoderBitRateStrategyKey encoder setting.
Notes: Available in OS X v10.9 and later.

4.53.383 AVSampleRateConverterAudioQualityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sample rate converter audio quality settings.
Notes: Value is an integer value. See AVAudioQuality* constants.

4.53.384 AVSampleRateKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the Audio setting keys that apply to all audio formats handled by the AVAudioPlayer and AVAudioRecorder classes.
Notes: A sample rate, in hertz, expressed as an floating point value.

4.53.385 AVStreamingKeyDeliveryContentKeyType as string

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

Function: A UTI for streaming key delivery content keys.
Notes: The value of this UTI is "com.apple.streamingkeydelivery.contentkey".

4.53.386 AVStreamingKeyDeliveryPersistentContentKeyType as string

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

Function: A UTI for persistent streaming key delivery content keys
Notes: The value of this UTI is "com.apple.streamingkeydelivery.persistentcontentkey".

4.53.387 AVTrackAssociationTypeAudioFallback as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Constants for identifying how other tracks are associated with a track.
Notes: An association from one audio track to another audio track. This constant indicates when one track contains the same content as another but in a more widely supported format. A player that does not support the format of the original track can use the "fallback" track instead.
For example, an asset may contain both stereo and a 5.1-channel audio tracks. In this case, marking the stereo track as the fallback for the 5.1-channel track would ensure that devices not capable of playing 5.1-channel audio can still play an equivalent track.
Available in OS X v10.9 and later.

4.53.388 AVTrackAssociationTypeChapterList as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Constants for identifying how other tracks are associated with a track.
Notes: An association from a track to another track containing chapter information, which may be a text track, a video track, or a timed metadata track.
Available in OS X v10.9 and later.

4.53.389 AVTrackAssociationTypeForcedSubtitlesOnly as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Constants for identifying how other tracks are associated with a track.
Notes: An association from one subtitle track, containing both forced and nonforced subtitles, to another subtitle track containing only forced subtitles for the same content.
Nonforced subtitles usually transcribe all ”normal” dialogue in a media asset and are typically not presented by default. Forced subtitles are those that are essential for presentation even when normal subtitles are disabled (for example, when a character speaks in a language foreign to that of the audio track).
Available in OS X v10.9 and later.

4.53.390 AVTrackAssociationTypeSelectionFollower as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Constants for identifying how other tracks are associated with a track.
Notes: An association from one track to another specifying that when the user selects the first track, the second should be considered an appropriate default selection.
For example, a follower for an audio track could be a subtitle track in the same language. When the user chooses a language for audio, the subtitle language ”follows” the audio language selection.
Available in OS X v10.9 and later.

4.53.391 AVTrackAssociationTypeTimecode as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Constants for identifying how other tracks are associated with a track.
Notes: An association from one track (of any type) to a timecode track containing timing information for the original track.
Available in OS X v10.9 and later.

4.53.392 AVURLAssetHTTPCookiesKey as string

Plugin Version: 14.4, Platform: macOS, Targets: All.

Function: HTTP cookies that the AVURLAsset may send with HTTP requests.
Notes: Standard cross-site policy still applies: cookies will only be sent to domains to which they apply.

By default, an AVURLAsset will only have access to cookies in the client's default cookie storage that apply to the AVURLAsset's URL. You can supplement the cookies available to the asset via use of this initialization option

HTTP cookies do not apply to non-HTTP(S) URLS.
In HLS, many HTTP requests (e.g., media, crypt key, variant index) might be issued to different paths or

hosts.
In both of these cases, HTTP requests will be missing any cookies that do not apply to the AVURLAsset's URL.
This init option allows the AVURLAsset to use additional HTTP cookies for those HTTP(S) requests.

4.53.393 AVURLAssetPreferPreciseDurationAndTimingKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys for AVAsset creations.
Example:

```
dim options as new Dictionary
options.value(AVFoundationMBS.AVURLAssetPreferPreciseDurationAndTimingKey) = true
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim a as AVAssetMBS = AVURLAssetMBS.URLAssetWithFile(f, options)

MsgBox str(a.duration.Seconds)+" seconds with precision: "+str(a.providesPreciseDurationAndTiming)
```

Notes: The corresponding value is a boolean, contained in an NSValue object, that indicates whether the asset should be prepared to indicate a precise duration and provide precise random access by time.
True indicates that longer loading times are acceptable in cases in which precise timing is required. Such precision, however, may require additional parsing of the resource in advance of operations that make use of any portion of it, depending on the specifics of its container format.
Many container formats provide sufficient summary information for precise timing and do not require additional parsing to prepare for it; QuickTime movie files and MPEG-4 files are examples of such formats. Other formats do not provide sufficient summary information, and precise random access for them is possible only after a preliminary examination of a file's contents.
If you only intend that the asset be played, the default value of false will suffice (because AVPlayer supports approximate random access by time when full precision isn't available). If you intend to insert the asset into an AVMutableComposition object, precise random access is typically desirable, and the value of true is recommended.

4.53.394 AVURLAssetReferenceRestrictionsKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the keys for AVAsset creations.
Notes: The corresponding value is a number wrapping an "AVAssetReferenceRestrictions" enum value—or the logical combination of multiple such values—that indicates the restrictions used by the asset when resolving references to external media data.
Some assets can contain references to media data stored outside the asset's container file, for example in another file. This key can be used to specify a policy to use when these references are encountered. If an

asset contains one or more references of a type that is forbidden by the reference restrictions, loading of asset properties will fail. In addition, such an asset cannot be used with other AVFoundation modules, such as AVPlayerItem or AVAssetExportSession.

4.53.395 AVVideoAverageBitRateKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the average bit rate (as bits per second) used in encoding.
The corresponding value is a number.
Apple documentation says H.264 only.

4.53.396 AVVideoCleanApertureHeightKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the clean aperture height.
The corresponding value is a number.

4.53.397 AVVideoCleanApertureHorizontalOffsetKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the clean aperture horizontal offset.
The corresponding value is a number.

4.53.398 AVVideoCleanApertureKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the clean aperture.
The corresponding value is a dictionary.

4.53.399 AVVideoCleanApertureVerticalOffsetKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the clean aperture vertical offset.
The corresponding value is a number.

4.53.400 AVVideoCleanApertureWidthKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the clean aperture width.
The corresponding value is a number.

4.53.401 AVVideoCodecAppleProRes422 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies that the video was encoded using the ProRes 422 Standard Definition encoder.
Files are identified with the .apcn extension.

4.53.402 AVVideoCodecAppleProRes4444 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies that the video was encoded using the Apple ProRes 4444 encoder.
Files are identified with the .ap4h extension.

4.53.403 AVVideoCodecH264 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies that the video was encoded using H264.

4.53.404 AVVideoCodecJPEG as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies that the video was encoded using the JPEG encoder.

4.53.405 AVVideoCodecKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the name of the codec used to encode the video.
The corresponding value is a string; equivalent to CMVideoCodecType.

4.53.406 AVVideoCodecTypeAppleProRes422 as string

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: One of the values used to describe codecs for video capture.
Notes: The Apple ProRes422 video codec.

4.53.407 AVVideoCodecTypeAppleProRes4444 as string

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: One of the values used to describe codecs for video capture.
Notes: The Apple ProRes4444 video codec.

4.53.408 AVVideoCodecTypeH264 as string

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: One of the values used to describe codecs for video capture.
Notes: The H.264 video codec.

4.53.409 AVVideoCodecTypeHEVC as string

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: One of the values used to describe codecs for video capture.
Notes: The HEVC video codec.

4.53.410 AVVideoCodecTypeJPEG as string

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: One of the values used to describe codecs for video capture.
Notes: The JPEG video codec.

4.53.411 AVVideoColorPrimariesKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key to identify color primaries in a color properties dictionary.

4.53.412 AVVideoColorPrimaries_EBU_3213 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoColorPrimariesKey.

4.53.413 AVVideoColorPrimaries_ITU_R_709_2 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoColorPrimariesKey.

4.53.414 AVVideoColorPrimaries_SMPTE_C as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoColorPrimariesKey.

4.53.415 AVVideoColorPropertiesKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key for a dictionary that contains properties specifying video color.
Notes: The dictionary must contain the keys AVVideoColorPrimariesKey, AVVideoTransferFunctionKey, and AVVideoYCbCrMatrixKey.

4.53.416 AVVideoCompressionPropertiesKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the compression properties.
The corresponding value is a Dictionary.

4.53.417 AVVideoHeightKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the height of the video in pixels.
The corresponding value is a number.

4.53.418 AVVideoMaxKeyFrameIntervalDurationKey as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Number (1 means key frames only, H.264 only)

4.53.419 AVVideoMaxKeyFrameIntervalKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the maximum interval between key frames.

The corresponding value is a number. 1 means key frames only.

4.53.420 AVVideoPixelAspectRatioHorizontalSpacingKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the pixel aspect ratio horizontal spacing.
The corresponding value is a number.

4.53.421 AVVideoPixelAspectRatioKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the pixel aspect ratio.
The corresponding value is a dictionary.

4.53.422 AVVideoPixelAspectRatioVerticalSpacingKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the pixel aspect ratio vertical spacing.
The corresponding value is a number.

4.53.423 AVVideoProfileLevelH264Baseline30 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a baseline level 3.0 profile.
Available in OS X v10.8 and later.

4.53.424 AVVideoProfileLevelH264Baseline31 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a baseline level 3.1 profile.

Available in OS X v10.8 and later.

4.53.425 AVVideoProfileLevelH264Baseline41 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a baseline level 4.1 profile.
Available in OS X v10.8 and later.

4.53.426 AVVideoProfileLevelH264BaselineAutoLevel as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Baseline Profile Auto Level.
Available on Mac OS X 10.9.

4.53.427 AVVideoProfileLevelH264High40 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: High Profile Level 4.0
Available on Mac OS X 10.9.

4.53.428 AVVideoProfileLevelH264High41 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: High Profile Level 4.1
Available on Mac OS X 10.9.

4.53.429 AVVideoProfileLevelH264HighAutoLevel as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: High Profile Auto Level

Available on Mac OS X 10.9.

4.53.430 AVVideoProfileLevelH264Main30 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a main level 3.0 profile.
Available in OS X v10.8 and later.

4.53.431 AVVideoProfileLevelH264Main31 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a main level 3.1 profile.
Available in OS X v10.8 and later.

4.53.432 AVVideoProfileLevelH264Main32 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a main level 3.2 profile.
Available in OS X v10.8 and later.

4.53.433 AVVideoProfileLevelH264Main41 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a main level 4.2 profile.
Available in OS X v10.8 and later.

4.53.434 AVVideoProfileLevelH264MainAutoLevel as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Main Profile Auto Level

Available on Mac OS X 10.9.

4.53.435 AVVideoProfileLevelKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the video profile.
Available in OS X v10.8 and later.

4.53.436 AVVideoQualityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the JPEG coded quality.
The corresponding value is a number 0.0-1.0.

JPEG and HEIC only. With HEIC, 1.0 indicates lossless compression

4.53.437 AVVideoScalingModeFit as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify how video should be scaled to fit a given area.
Notes: Crop to remove edge processing region; preserve aspect ratio of cropped source by reducing specified width or height if necessary.
This mode does not scale a small source up to larger dimensions.

4.53.438 AVVideoScalingModeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify how video should be scaled to fit a given area.
Notes: A key to retrieve the video scaling mode from a dictionary.

4.53.439 AVVideoScalingModeResize as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify how video should be scaled to fit a given area.
Notes: Crop to remove edge processing region; scale remainder to destination area.
This mode does not preserve the aspect ratio.

4.53.440 AVVideoScalingModeResizeAspect as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify how video should be scaled to fit a given area.
Notes: Preserve aspect ratio of the source, and fill remaining areas with black to fit destination dimensions.

4.53.441 AVVideoScalingModeResizeAspectFill as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants to specify how video should be scaled to fit a given area.
Notes: Preserve aspect ratio of the source, and crop picture to fit destination dimensions.

4.53.442 AVVideoTransferFunctionKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key to identify the transfer function in a color properties dictionary.

4.53.443 AVVideoTransferFunction_ITU_R_709_2 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoTransferFunctionKey.

4.53.444 AVVideoTransferFunction_SMPTE_240M_1995 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoTransferFunctionKey.

4.53.445 AVVideoWidthKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the constants for the dictionary with video settings.
Notes: Specifies a key to access the width of the video in pixels.
The corresponding value is a number.

4.53.446 AVVideoYCbCrMatrixKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key to identify the Y'CbCr matrix in a color properties dictionary.

4.53.447 AVVideoYCbCrMatrix_ITU_R_601_4 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoYCbCrMatrixKey.

4.53.448 AVVideoYCbCrMatrix_ITU_R_709_2 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoYCbCrMatrixKey.

4.53.449 AVVideoYCbCrMatrix_SMPTE_240M_1995 as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the values to use with AVVideoYCbCrMatrixKey.

4.53.450 WriteCGImageToFile(File as FolderItem, Type as String, Image as Variant, options as Dictionary = nil, tag as Variant = nil) as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Writes a CGImage to a file on disk on another thread.
Notes: Image must be a CGImageMBS.

If this method returns true (parameters are correct), it starts the compression and writing of the image on a helper thread and later calls WriteCGImageToFileCompleted event to inform you about success or failure. Returns false in case of bad parameters.

Tag is the value you passed when calling the original method.

4.53.451 Events

4.53.452 AssetLoadValuesAsynchronouslyForKeysFinished(MetadataItem as AVMetadataItemMBS, keys() as string, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by AVAssetMBS.loadValuesAsynchronouslyForKeys method on completion.
Notes: Tag is the value you passed when calling the original method.

4.53.453 AssetTrackLoadValuesAsynchronouslyForKeysFinished(MetadataItem as AVMetadataItemMBS, keys() as string, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by AVAssetTrackMBS.loadValuesAsynchronouslyForKeys method on completion.
Notes: Tag is the value you passed when calling the original method.

4.53.454 audioPlayerDecodeErrorDidOccur(player as AVAudioPlayerMBS, error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when an audio player encounters a decoding error during playback.
Notes: player: The audio player that encountered the decoding error.
error: The decoding error.

4.53.455 audioPlayerDidFinishPlaying(player as AVAudioPlayerMBS, successful as boolean)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when a sound has finished playing.
Notes: player: The audio player that finished playing.

flag: True on successful completion of playback; false if playback stopped because the system could not decode the audio data.

This method is not called upon an audio interruption. Rather, an audio player is paused upon interruption—the sound has not finished playing.

4.53.456 audioRecorderDidFinishRecording(recorder as AVAudioRecorderMBS, successful as boolean)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by the system when a recording is stopped or has finished due to reaching its time limit.
Notes: recorder: The audio recorder that has finished recording.
flag: True on successful completion of recording; False if recording stopped because of an audio encoding error.

This method is not called by the system if the audio recorder stopped due to an interruption.

4.53.457 audioRecorderEncodeErrorDidOccur(recorder as AVAudioRecorderMBS, error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when an audio recorder encounters an encoding error during recording.
Notes: recorder: The audio recorder that encountered the encoding error.
error: The encoding error.

4.53.458 BoundaryTimeObserver(Player as AVPlayerMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by addBoundaryTimeObserverForTimes.
Notes: Tag is the value you passed when calling the original method.

4.53.459 CaptureDeviceSubjectAreaDidChange(device as AVCaptureDeviceMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Notification that is posted when the instance of AVCaptureDevice has detected a substantial change to the video subject area.
Notes: notification is a NSNotificationMBS object.
This notification is only sent if you first set subjectAreaChangeMonitoringEnabled to True.

4.53.460 CaptureDeviceWasConnected(device as AVCaptureDeviceMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Notification that is posted when a new device becomes available.
Notes: notification is a NSNotificationMBS object.

4.53.461 CaptureDeviceWasDisconnected(device as AVCaptureDeviceMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Notification that is posted when an existing device becomes unavailable.
Notes: notification is a NSNotificationMBS object.

4.53.462 CaptureInputPortFormatDescriptionDidChange(InputPort as AVCaptureInputPortMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted if the formatDescription of the capture input port changes.
Notes: notification is a NSNotificationMBS object.

4.53.463 captureOutputDidDropSampleBuffer(captureOutput as AVCaptureOutputMBS, OutputSampleBuffer as CMSampleBufferMBS, connection as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Notifies the delegate that a video frame was discarded.
Notes: captureOutput: The capture output object.
sampleBuffer: A CMSampleBuffer object containing information about the dropped frame, such as its format and presentation time. This sample buffer contains none of the original video data.
connection: The connection from which the video was received.

Delegates receive this message whenever a late video frame is dropped. This method is called once for each dropped frame. It is called on the main thread, so please make your event code run very fast.

4.53.464 captureOutputDidFinishRecordingToOutputFileAtURL(captureOutput as AVCaptureFileOutputMBS, outputFileURL as string, connections() as AVCaptureConnectionMBS, error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Informs the delegate when all pending data has been written to an output file.
Notes: captureOutput: The capture file output that has finished writing the file.
outputFileURL: The file URL of the file that is being written.
connections: An array of AVCaptureConnection objects attached to the file output that provided the data that is being written to the file.
error: If the file was not written successfully, an error object that describes the problem; otherwise nil.

This method is called whenever a file is finished. If the file was forced to be finished due to an error, the error is described in the error parameter—otherwise, the error parameter is nil.

This method is called when the file output has finished writing all data to a file whose recording was stopped, either because startRecordingToOutputFileURL or stopRecording were called, or because an error (described by the error parameter), occurred (if no error occurred, the error parameter is nil).

This method is always called for each recording request, even if no data is successfully written to the file.

4.53.465 captureOutputDidOutputMetadataObjects(captureOutput as AVCaptureOutputMBS, metadataObjects() as AVMetadataObjectMBS, connection as AVCaptureConnectionMBS)

Plugin Version: 14.0, Platform: iOS, Targets: .

Function: Called by AVCaptureMetadataOutputMBS class if new metadata is available.
Notes: Only for iOS!

captureOutput: The AVCaptureMetadataOutputMBS object that captured and emitted the metadata objects.
metadataObjects: An array of AVMetadataObjectMBS instances representing the newly emitted metadata. Because AVMetadataObject is an abstract class, the objects in this array are always instances of a concrete subclass.
connection: The capture connection through which the objects were emitted.

The AVCaptureMetadataOutputMBS object emits only metadata objects whose types are included in its metadataObjectTypes property. The delegate implements this method to perform additional processing on metadata objects as they become available. If you plan to use metadata objects outside the scope of this method, you must store strong references to them and remove those references when the objects are no longer required.

4.53.466 captureOutputDidOutputSampleBuffer(captureOutput as AVCaptureOutputMBS, OutputSampleBuffer as CMSampleBufferMBS, connection as AVCaptureConnectionMBS, features as variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called for frames being outputted.
Notes: Called by AVCaptureFileOutputMBS, AVCaptureVideoDataOutputMBS and AVCaptureAudioDataOutputMBS.

features is an array of object, which contains detected CIFeatureMBS objects for AVCaptureVideoDataOutputMBS with CIDetector.

4.53.467 captureOutputDidPauseRecordingToOutputFileAtURL(captureOutput as AVCaptureFileOutputMBS, fileURL as string, connections() as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called whenever the output is recording to a file and successfully pauses the recording at the request of a client.
Notes: captureOutput: The capture file output that has paused its file recording.
fileURL: The file URL of the file that is being written.
connections: An array of AVCaptureConnection objects attached to the file output that provided the data that is being written to the file.

This method is called whenever a request to pause recording is actually respected.

It is safe for delegates to change what the file output is currently doing (starting a new file, for example) from within this method. If recording to a file is stopped, either manually or due to an error, this method is not guaranteed to be called, even if a previous call to pauseRecording was made.

4.53.468 captureOutputDidResumeRecordingToOutputFileAtURL(captureOutput as AVCaptureFileOutputMBS, fileURL as string, connections() as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called whenever the output, at the request of the client, successfully resumes a file recording that was paused.
Notes: captureOutput: The capture file output that has resumed its paused file recording.
fileURL: The file URL of the file that is being written.
connections: An array of AVCaptureConnection objects attached to the file output that provided the data that is being written to the file.

Delegates can use this method to be informed when a request to resume recording is actually respected.

It is safe for delegates to change what the file output is currently doing (starting a new file, for example) from within this method. If recording to a file is stopped, either manually or due to an error, this method is not guaranteed to be called, even if a previous call toresumeRecording was made.

4.53.469 captureOutputDidStartRecordingToOutputFileAtURL(captureOutput as AVCaptureFileOutputMBS, fileURL as string, connections() as AVCaptureConnectionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Informs the delegate when the output has started writing to a file.
Notes: captureOutput: The capture file output that started writing the file.
fileURL: The file URL of the file that is being written.
connections: An array of AVCaptureConnection objects attached to the file output that provided the data that is being written to the file.

If an error condition prevents any data from being written, this method may not be called. captureOutputwillFinishRecordingToOutputFileAtURL and captureOutputdidFinishRecordingToOutputFileAtURL are always called, even if no data is written.

4.53.470 captureOutputWillFinishRecordingToOutputFileAtURL(captureOutput as AVCaptureFileOutputMBS, fileURL as string, connections() as AVCaptureConnectionMBS, error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Informs the delegate when the output will stop writing new samples to a file.

Notes: captureOutput: The capture file output that will finish writing the file.
fileURL: The file URL of the file that is being written.
connections: An array of AVCaptureConnection objects attached to the file output that provided the data that is being written to the file.
error: An error describing what caused the file to stop recording, or nil if there was no error.

This method is called when the file output will stop recording new samples to the file at outputFileURL, either because startRecordingToOutputFileURL or stopRecording were called, or because an error (described by the error parameter), occurred (if no error occurred, the error parameter is nil).

This method is always called for each recording request, even if no data is successfully written to the file.

4.53.471 CaptureSessionDidStartRunning(session as AVCaptureSessionMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted when a capture session starts.
Notes: notification is a NSNotificationMBS object.

4.53.472 CaptureSessionDidStopRunning(session as AVCaptureSessionMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted when a capture session stops.
Notes: notification is a NSNotificationMBS object.

4.53.473 CaptureSessionRuntimeError(session as AVCaptureSessionMBS, error as NSErrorMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted if an error occurred during a capture session.
Notes: notification is a NSNotificationMBS object.
You retrieve the underlying error from the notification's user info dictionary using the key AVCaptureSessionErrorKey.

4.53.474 captureStillImageAsynchronouslyCompleted(CaptureStillImageOutput as AVCaptureStillImageOutputMBS, prepareJpegStillImage as boolean, tag as Variant, error as NSErrorMBS, imageDataSampleBuffer as CMSampleBufferMBS, JpegStillImage as memoryblock)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: The event called when still image was captured.
Notes: If prepareJpegStillImage is true, the plugin will prepare the jpeg data and pass it as JpegStillImage, so the preparation is done on the capture thread.
imageDataSampleBuffer: The data that was captured.

The buffer attachments may contain metadata appropriate to the image data format. For example, a buffer containing JPEG data may carry a kCGImagePropertyExifDictionary as an attachment. See ImageIO/CGImageProperties.h for a list of keys and value types.
error
If the request could not be completed, an NSError object that describes the problem; otherwise nil.

Tag is the value you passed when calling the original method.

4.53.475 determineCompatibilityOfExportPresetCompleted(presetName as string, asset as AVAssetMBS, outputFileType as string, compatible as boolean, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The event called by AVAssetExportSessionMBS.determineCompatibilityOfExportPreset.
Notes: The event to execute with the results of your call to determineCompatibilityOfExportPreset.
The compatible parameter contains the value true if the combination of options is compatible or false if they are not.

Tag is the value you passed when calling the original method.

4.53.476 determineCompatibleFileTypesCompleted(exportsession as AVAssetExportSessionMBS, compatibleFileTypes() as string, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The event called when you call AVAssetExportSessionMBS.determineCompatibleFileTypes method.
Notes: The event to execute with the results.
You receive an array parameter and returns no result. The array contains zero or more strings with the UTIs of the compatible file types. The file types are in no particular order. For a list of constants specifying

UTIs for standard file types, see AV Foundation Constants Reference.

Tag is the value you passed when calling the original method.

4.53.477 exportAsynchronouslyCompleted(ExportSession as AVAssetExportSessionMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Export finished.
Notes: Called by AVAssetExportSessionMBS.exportAsynchronously when the export is done or failed.

Tag is the value you passed when calling the original method.

4.53.478 finishWritingCompleted(writer as AVAssetWriterMBS, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: Event called from AVAssetWriterMBS.finishWritingWithCompletionHandler.
Notes: The event is invoked once the writing of the output file is finished or if a failure or cancellation occurs in the meantime.

Tag is the value you passed when calling the original method.

4.53.479 generateCGImagesAsynchronouslyForTimesCompleted(generator as AVAssetImageGeneratorMBS, requestedTime as CMTimeMBS, image as Variant, actualTime as CMTimeMBS, result as Integer, error as NSErrorMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when image generation is completed or cancelled.
Notes: Image is a CGImageMBS object.
Called by AVAssetImageGeneratorMBS.generateCGImagesAsynchronouslyForTimes method.

Tag is the value you passed when calling the original method.

4.53.480 legibleOutputDidOutputAttributedStrings(output as AVPlayerItemLegibleOutputMBS, strings() as Variant, nativeSamples() as CMSampleBufferMBS, itemTime as CMTimeMBS)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The legible output did output an attributed string.
Notes: Strings array contains NSAttributedStringMBS objects.

4.53.481 MetadataItemLoadValuesAsynchronouslyForKeysFinished(MetadataItem as AVMetadataItemMBS, keys() as string, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by AVMetadataItemMBS.loadValuesAsynchronouslyForKeys method on completion.
Notes: Tag is the value you passed when calling the original method.

4.53.482 outputMediaDataWillChange(output as AVPlayerItemOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Tells you that new samples are about to arrive.
Notes: You can use this method to prepare for any new sample data. This method is called at some point after a call to your video output object's requestNotificationOfMediaDataChangeWithAdvanceInterval method.

4.53.483 outputSequenceWasFlushed(output as AVPlayerItemOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Tells the delegate that a new sample sequence is commencing.
Notes: output: The output object that sent the message.

This method is called after any attempt to seek or change the playback direction of the item's content. If you are maintaining any queued future samples, you can use your implementation of this method to discard those samples.

4.53.484 PeriodicTimeObserver(Player as AVPlayerMBS, time as CMTimeMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by addPeriodicTimeObserverForInterval.
Notes: time: The time at which the block is invoked.

Tag is the value you passed when calling the original method.

4.53.485 PlayerItemDidPlayToEndTime(PlayerItem as AVPlayerItemMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted when the item has played to its end time.
Notes: notification is a NSNotificationMBS object.
The notification's object is the item that finished playing.

4.53.486 PlayerItemFailedToPlayToEndTime(PlayerItem as AVPlayerItemMBS, error as NSErrorMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted when the item failed to play to its end time.
Notes: notification is a NSNotificationMBS object.
The notification's object is the item that finished playing.
The user info dictionary contains an error object that describes the problem—seeAVPlayerItemFailedToPlayToEndTimeErrorKey.

4.53.487 PlayerItemNewAccessLogEntry(PlayerItem as AVPlayerItemMBS, notification as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: A new access log entry is available to a player item.
Notes: notification is a NSNotificationMBS object.
Posted when a new access log entry has been added.
The notification's object is the player item. The new log entry is accessible via accessLog.

4.53.488 PlayerItemNewErrorLogEntry(PlayerItem as AVPlayerItemMBS, notification as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: A new error log entry is available to a player item.
Notes: notification is a NSNotificationMBS object.
Posted when a new error log entry has been added.
The notification's object is the player item. The new log entry is accessible via errorLog, respectively.

4.53.489 PlayerItemPlaybackStalled(PlayerItem as AVPlayerItemMBS, notification as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The player item playback is stalled.
Notes: notification is a NSNotificationMBS object.

Posted when some media did not arrive in time to continue playback.
The notification's object is the AVPlayerItem instance whose playback was unable to continue because the necessary media isn't available yet, usually because it didn't arrive in a timely fashion from its source over a network. Playback will continue once a sufficient amount of media has subsequently been delivered.

4.53.490 playerItemSeekToDateFinished(player as AVPlayerItemMBS, date as Variant, finished as boolean, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The seek operation to date finished.
Notes: Tag is the value you passed when calling the original method.

4.53.491 playerItemSeekToTimeFinished(player as AVPlayerItemMBS, time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, finished as boolean, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: A seek operation has finished.
Notes: finished: Indicates whether the seek operation completed.

Tag is the value you passed when calling the original method.

4.53.492 PlayerItemTimeJumped(PlayerItem as AVPlayerItemMBS, notification as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Posted when the item's current time has changed discontinuously.
Notes: notification is a NSNotificationMBS object.
The notification's object is the item.

4.53.493 playerSeekToDateFinished(player as AVPlayerMBS, date as Variant, finished as boolean, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: Event called by AVPlayerMBS.seekToDate method when seeking is done.
Notes: Tag is the value you passed when calling the original method.

4.53.494 playerSeekToTimeFinished(player as AVPlayerMBS, time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, finished as boolean, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called by AVPlayerMBS.seekToTime when seeking is done.
Notes: finished: Indicates whether the seek operation completed.

Tag is the value you passed when calling the original method.

4.53.495 prerollAtRateFinished(player as AVAudioPlayerMBS, rate as Double, finished as boolean, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: The event called when preroll is done.
Notes: Finished: True if the data was loaded or false if there was a problem.
The value might be false if the preroll was interrupted by a time change or incompatible rate change.

Tag is the value you passed when calling the original method.

4.53.496 requestAccessForMediaTypeCompleted(MediaType as String, granted as boolean, tag as variant)

Plugin Version: 18.4, Platform: macOS, Targets: .

Function: The call to AVCaptureDeviceMBS.requestAccessForMediaType was decided.

4.53.497 requestContentAuthorizationCompleted(PlayerItem as AVPlayerItemMBS, timeoutInterval as Double, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: The event called when a request started with AVPlayerItemMBS.requestContentAuthorizationAsynchronously completed.
Notes: Tag is the value you passed when calling the original method.

4.53.498 requestMediaDataWhenReadyOnQueueCompleted(assetWriterInput as AVAssetWriterInputMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when media is ready.
Notes: You called requestMediaDataWhenReadyOnQueue on an AVAssetWriterInputMBS before.

Tag is the value you passed when calling the original method.

4.53.499 requestMediaDataWhenReadyOnQueueFinished(assetWriterInput as AVAssetWriterInputMBS, assetReaderOutput as AVAssetReaderOutputMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when copying data from input to output by plugin is done.
Notes: Tag is the value you passed when calling the original method.

4.53.500 requestMediaDataWhenReadyOnQueueProgress(assetWriterInput as AVAssetWriterInputMBS, assetReaderOutput as AVAssetReaderOutputMBS, convertedByteCount as Int64, LastBuffer as CMSampleBufferMBS, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Called when media is processed from input to output from plugin.
Notes: LastBuffer: The last buffer processed.

Tag is the value you passed when calling the original method.

4.53.501 resourceLoaderDidCancelLoadingRequest(resourceLoader as AVAssetResourceLoaderMBS, loadingRequest as AVAssetResourceLoadingRequestMBS)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: Invoked to inform the delegate that a prior loading request has been cancelled.
Notes: resourceLoader: The resource loader.
loadingRequest: The loading request that has been cancelled.

Previously issued loading requests can be cancelled when data from the resource is no longer required or when a loading request is superseded by new requests for data from the same resource.

For example, if to complete a seek operation it becomes necessary to load a range of bytes that's different from a range previously requested, the prior request may be cancelled while the delegate is still handling it. Available in OS X v10.9 and later.

4.53.502 resourceLoaderShouldWaitForLoadingOfRequestedResource(resourceLoader as AVAssetResourceLoaderMBS, loadingRequest as AVAssetResourceLoadingRequestMBS) as boolean

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: Asks the delegate if it wants to load the requested resource.
Notes: resourceLoader: The resource loader object that is making the request.
loadingRequest: The loading request object that contains information about the requested resource.

Returns true if your delegate can load the resource specified by the loadingRequest parameter or false if it cannot.

The resource loader object calls this method when assistance is required of your code to load the specified resource. For example, the resource loader might call this method to load decryption keys that have been specified using a custom URL scheme.

Returning true from this method, implies only that the receiver will load, or at least attempt to load, the resource. In some implementations, the actual work of loading the resource might be initiated on another thread, running asynchronously to the resource loading delegate; whether the work begins immediately or merely soon is an implementation detail of the client application.

You can load the resource synchronously or asynchronously. In both cases, you must indicate success or failure of the operation by calling the finishLoadingWithResponse or finishLoadingWithError: method of the request object when you finish. If you load the resource asynchronously, you must also store a strong reference to the object in the loadingRequest parameter before returning from this method.

If you return false from this method, the resource loader treats the loading of the resource as having failed. Available in OS X v10.9 and later.

4.53.503 SampleBufferDisplayLayerMediaDataWhenReady(tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Event called by AVSampleBufferDisplayLayerMBS.requestMediaDataWhenReady when media data is ready.
Notes: Tag is the value you passed when calling the original method.

4.53.504 videoCompositionShouldContinueValidatingAfterFindingEmptyTimeRange(videoComposition as AVVideoCompositionMBS, timeRange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Reports a time range that has no corresponding video composition instruction.
Notes: videoComposition: The video composition being validated.
timeRange: The time range that has no corresponding video composition instruction.

Return true if the video composition should continue validation in order to report additional problems that may exist, otherwise false.

4.53.505 videoCompositionShouldContinueValidatingAfterFindingInvalidTimeRangeInInstruction(videoComposition as AVVideoCompositionMBS, Instruction as AVVideoCompositionInstructionMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Reports a video composition instruction with a time range that is invalid, that overlaps with the time range of a prior instruction, or that contains times earlier than the time range of a prior instruction.
Notes: videoComposition: The video composition being validated.
videoCompositionInstruction: The video composition instruction.
Return true if the video composition should continue validation in order to report additional problems that may exist, otherwise false.

4.53.506 videoCompositionShouldContinueValidatingAfterFindingInvalidTrackIDInInstruction(videoComposition as AVVideoCompositionMBS, videoCompositionInstruction as AVVideoCompositionInstructionMBS, layerInstruction as AVVideoCompositionLayerInstructionMBS, asset as AVAssetMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Reports a video composition layer instruction with a track ID that does not correspond either to the track ID used for the composition's animation tool or to a track of the asset specified in isValidForAsset.
Notes: videoComposition: The video composition being validated.
videoCompositionInstruction: The video composition instruction.
layerInstruction: The layer instruction.
asset: The underlying asset.

Return true if the video composition should continue validation in order to report additional problems that may exist, otherwise false.

4.53.507 videoCompositionShouldContinueValidatingAfterFindingInvalidValueForKey(videoComposition as AVVideoCompositionMBS, key as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: .

Function: Reports that a key that has an invalid value.
Notes: videoComposition: The video composition being validated.
key: The key being validated.

Return true if the video composition should continue validation in order to report additional problems that may exist, otherwise false.

4.53.508 WriteCGImageToFileCompleted(file as folderitem, type as string, image as Variant, options as dictionary, success as boolean, tag as Variant)

Plugin Version: 14.0, Platform: macOS, Targets: .

Function: The call to WriteCGImageToFile method finished.
Notes: Image is a CGImageMBS.
Success property tells you whether or not the write was successful.

Tag is the value you passed when calling the original method.

4.53.509 Constants

Constants

Constant	Value	Description
kAudioFormat60958AC3	”cac3”	One of the audio formats. A key that specifies an AC-3 codec that provides data package over an IEC 60958 compliant digital audio interface. Uses the st flags in ”AudioStreamBasicDescription Flags”.
kAudioFormatAC3	”ac-3”	One of the audio formats. A key that specifies an AC-3 codec. Uses no flags.
kAudioFormatAES3	”aes3”	One of the audio formats. The format defined by the AES3-2003 standard. Adopted i MPEG-2 containers and SDTI transport streams with SMPT 2002 and 331M-2000. Uses no flags.
kAudioFormatALaw	”alaw”	One of the audio formats. aLaw 2:1. Uses no flags.
kAudioFormatAMR	”samr”	One of the audio formats. The AMR (Adaptive Multi-Rate) narrow band speech codec.
kAudioFormatAppleIMA4	”ima4”	One of the audio formats. A key that specifies Apple’s implementation of the IMA 4:1 A Uses no flags.
kAudioFormatAppleLossless	”alac”	One of the audio formats. Apple Lossless. Uses no flags.
kAudioFormatAudible	”AUDB”	One of the audio formats. The codec used for Audible, Inc. audio books. Uses no flags.
kAudioFormatDVIIntelIMA	&h6D730011	One of the audio formats. DVI/Intel IMA ADPCM - ACM code 17.
kAudioFormatiLBC	”ilbc”	One of the audio formats. The iLBC (internet Low Bitrate Codec) narrow band speech c flags.
kAudioFormatLinearPCM	”lpcm”	One of the audio formats. A key that specifies linear PCM, a noncompressed audio data fo frame per packet. Uses the linear PCM format flags in ”AudioS scription Flags”.
kAudioFormatMACE3	”MAC3”	One of the audio formats. MACE 3:1. Uses no flags.
kAudioFormatMACE6	”MAC6”	One of the audio formats. MACE 6:1. Uses no flags.
kAudioFormatMicrosoftGSM	&h6D730031	One of the audio formats. Microsoft GSM 6.10 - ACM code 49.
kAudioFormatMIDIStream	”midi”	One of the audio formats. A stream of MIDIPacketList structures where the time stamp Packet structures are sample offsets in the stream. The mSa in the AudioStreamBasicDescription structure is used to descri passed in this kind of stream and an audio unit that receives or stream can use this sample rate together with the number of fr dering and the sample offsets within the MIDIPacketList to defi any MIDI event within this list. Uses no flags.
kAudioFormatMPEG4AAC	”aac ”	One of the audio formats. A key that specifies an MPEG-4 AAC codec. The flags fiel MPEG-4 audio object type constant listed in ”MPEG-4 Audi Constants” indicating the specific kind of data.
kAudioFormatMPEG4AAC_ELD	”aace”	One of the audio formats. MPEG-4 AAC Enhanced Low Delay audio object. Uses no flag
kAudioFormatMPEG4AAC_ELD_SBR	”aacf”	One of the audio formats. MPEG-4 AAC Enhanced Low Delay audio object with SBR replication) extension layer. Uses no flags.
kAudioFormatMPEG4AAC_ELD_V2	”aacg”	One of the audio formats.
kAudioFormatMPEG4AAC_HE	”aach”	One of the audio formats. MPEG-4 High Efficiency AAC audio object. Uses no flags.
kAudioFormatMPEG4AAC_HE_V2	”aacp”	One of the audio formats. MPEG-4 High Efficiency AAC Version 2 audio object. Uses no

Audio Quality Values

Constant	Value	Description
AVAudioQualityHigh	&h60	High quality sample rate conversion.
AVAudioQualityLow	&h20	Low quality rate conversion.
AVAudioQualityMax	&h7F	Maximum quality sample rate conversion.
AVAudioQualityMedium	&h40	Medium quality sample rate conversion.
AVAudioQualityMin	0	The minimum quality for sample rate conversion.

Error Constants

Constant	Value	Description
AVErrorApplicationIsNotAuthorized	-11836	The application is not authorized to play media.
AVErrorApplicationIsNotAuthorizedToUseDevice	-11852	The user has denied this application permission fo Available in OS X v10.9 and later.
AVErrorCompositionTrackSegmentsNotContiguous	-11824	The source media can't be added because it conta
AVErrorContentIsNotAuthorized	-11835	The user is not authorized to play the media.
AVErrorContentIsProtected	-11831	The application is not authorized to open the med
AVErrorDecodeFailed	-11821	The operation could not be completed because so be decoded.
AVErrorDecoderNotFound	-11833	The decoder for the given media was not found The error's userInfo may contain values for the k and AVErrorMediaSubTypeKey, if they are availa
AVErrorDecoderTemporarilyUnavailable	-11839	The appropriate decoder is currently not available The error's userInfo may contain AVErrorMediaT SubTypeKey, if they are available.
AVErrorDeviceAlreadyUsedByAnotherSession	-11804	Media could not be captured from the device beca where in this application.
AVErrorDeviceInUseByAnotherApplication	-11815	The device could not be opened because it is in u
AVErrorDeviceLockedForConfigurationByAnotherProcess	-11817	Settings for the device could not be changed beca trolled by another application.
AVErrorDeviceNotConnected	-11814	The device could not be opened because it is not
AVErrorDeviceWasDisconnected	-11808	Recording stopped because the device was turned
AVErrorDiskFull	-11807	Recording stopped because the disk is getting full
AVErrorDisplayWasDisabled	-11845	Screen capture failed because the display was inac
AVErrorEncoderNotFound	-11834	The requested encoder was not found. The error's userInfo may contain values for the k and AVErrorMediaSubTypeKey, if they are availa
AVErrorEncoderTemporarilyUnavailable	-11840	The appropriate encoder is currently not available The error's userInfo may contain AVErrorMediaT SubTypeKey, if they are available.
AVErrorExportFailed	-11820	The export could not be completed.
AVErrorFailedToLoadMediaData	-11849	The media resource does not contain all of the d loaded. Available in OS X v10.9 and later.
AVErrorFileAlreadyExists	-11823	The file could not be created because a file with t in the same location.
AVErrorFileFailedToParse	-11829	The media could not be opened because the file is nized format.
AVErrorFileFormatNotRecognized	-11828	The media could not be opened because it is not
AVErrorIncompatibleAsset	-11848	The media could not be displayed because the i playing the content. This might occur if the devic particular video profile level. Available in OS X v10.9 and later.
AVErrorInvalidCompositionTrackSegmentDuration	-11825	The source media can't be added because its du invalid.
AVErrorInvalidCompositionTrackSegmentSourceDuration	-11827	The source media can't be added because it has n
AVErrorInvalidCompositionTrackSegmentSourceStartTime	-11826	The source media can't be added because its sta invalid.
AVErrorInvalidOutputURLPathExtension	-11843	The path extension of the URL was invalid.
AVErrorInvalidSourceMedia	-11822	The operation could not be completed because so be read.
AVErrorInvalidVideoComposition	-11841	You attempted to perform an operation with the
AVErrorMaximumDurationReached	-11810	Recording stopped because the maximum duratio
AVErrorMaximumFileSizeReached	-11811	Recording stopped because the maximum size for
AVErrorMaximumNumberOfSamplesForFileFormatReached	-11813	Recording stopped because the maximum numbe reached.
AVErrorMaximumStillImageCaptureRequestsExceeded	-11830	The photo could not be taken because there are to haven't completed yet.
AVErrorMediaChanged	-11809	Recording stopped because the format of the sour

4.54 class AVFrameRateRangeMBS

4.54.1 class AVFrameRateRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVFrameRateRange object expresses a range of valid frame rates as minimum and maximum rate and minimum and maximum duration.
Notes: An AVFrameRateRange object is immutable.
An AVCaptureDeviceFormat object wraps a CMFormatDescription and expresses a range of valid video frame rates as an array of AVFrameRateRange objects.
An AVCaptureDevice object uses AVCaptureDeviceFormat to describe the formats it supports and the currently-active format.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.2pr4

4.54.2 Methods

4.54.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.54.4 DisplayName as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The display text for this object.
Notes: The plugin formats a string with the frame rate range for displaying.

4.54.5 maxFrameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The maximum frame duration supported by the range. (read-only)
Notes: This value is the reciprocal of minFrameRate, and expresses the minimum frame rate as a duration.

4.54.6 maxFrameRate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The maximum frame rate supported by the range. (read-only)
Notes: The frame is given in frames per second.

4.54.7 minFrameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The minimum frame duration supported by the range. (read-only)
Notes: This value is the reciprocal of maxFrameRate, and expresses the maximum frame rate as a duration.

4.54.8 minFrameRate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The minimum frame rate supported by the range. (read-only)
Notes: The frame is given in frames per second.

4.54.9 Properties

4.54.10 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.55 class AVMediaSelectionGroupMBS

4.55.1 class AVMediaSelectionGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMediaSelectionGroup provides a collection of mutually exclusive options for the presentation of media within an asset.

4.55.2 Methods

4.55.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.55.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.55.5 copy as AVMediaSelectionGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.55.6 mediaSelectionOptionsFromArrayFilteredAndSortedAccordingToPreferredLanguages(mediaSelectionOptions() as AVMediaSelectionOptionMBS) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to whether their locales match any language identifier in the specified array of preferred languages. The returned array is sorted according to the order of preference of the language each matches.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOptions to be filtered and sorted.

preferredLanguages: An array of language identifiers in order of preference, each of which is an IETF BCP 47 (RFC 4646) language identifier. Use NSLocaleMBS.preferredLanguages to obtain the user's list of preferred languages. Plugins passes NSLocaleMBS.preferredLanguages if you don't provide an array of languages.

Returns an instance of NSArray containing media selection options of the specified NSArray that match a preferred language, sorted according to the order of preference of the language each matches.
See also:

- 4.55.7 mediaSelectionOptionsFromArrayFilteredAndSortedAccordingToPreferredLanguages(mediaSelectionOptions() as AVMediaSelectionOptionMBS, preferredLanguages() as string) as AVMediaSelectionOptionMBS() 481

4.55.7 mediaSelectionOptionsFromArrayFilteredAndSortedAccordingToPreferredLanguages(mediaSelectionOptions() as AVMediaSelectionOptionMBS, preferredLanguages() as string) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to whether their locales match any language identifier in the specified array of preferred languages. The returned array is sorted according to the order of preference of the language each matches.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOptions to be filtered and sorted.
preferredLanguages: An array of language identifiers in order of preference, each of which is an IETF BCP 47 (RFC 4646) language identifier. Use NSLocaleMBS.preferredLanguages to obtain the user's list of preferred languages. Plugins passes NSLocaleMBS.preferredLanguages if you don't provide an array of languages.

Returns an instance of NSArray containing media selection options of the specified NSArray that match a preferred language, sorted according to the order of preference of the language each matches.
See also:

- 4.55.6 mediaSelectionOptionsFromArrayFilteredAndSortedAccordingToPreferredLanguages(mediaSelectionOptions() as AVMediaSelectionOptionMBS) as AVMediaSelectionOptionMBS() 480

4.55.8 mediaSelectionOptionsFromArrayWithLocale(mediaSelectionOptions() as AVMediaSelectionOptionMBS, locale as NSLocaleMBS) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to locale.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOption to be filtered by locale.
locale: The NSLocale that must be matched for a media selection option to be copied to the output array.
Returns an instance of Array containing the media selection options of the specified Array that match the specified locale.

4.55.9 mediaSelectionOptionsFromArrayWithMediaCharacteristics(mediaSelectionOptions() as AVMediaSelectionOptionMBS, mediaCharacteristics() as string) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to one or more media characteristics.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOptions to be filtered by media characteristic.
mediaCharacteristics: The media characteristics that must be matched for a media selection option to be copied to the output array.

Returns an instance of NSArray containing the media selection options of the specified NSArray that match the specified media characteristics.

4.55.10 mediaSelectionOptionsFromArrayWithoutMediaCharacteristics(mediaSelectionOptions() as AVMediaSelectionOptionMBS, mediaCharacteristics() as string) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to whether they lack one or more media characteristics.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOptions to be filtered by media characteristic.
mediaCharacteristics: The media characteristics that must not be present for a media selection option to be copied to the output array.
Returns an instance of NSArray containing the media selection options of the specified NSArray that lack the specified media characteristics.

4.55.11 mediaSelectionOptionWithPropertyList(plist as Variant) as AVMediaSelectionOptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the instance of AVMediaSelectionOption with properties that match the specified property list.
Notes: plist: A property list previously obtained from an option in the group via AVMediaSelectionOption.propertyList.
If the specified properties match those of an option in the group, an instance of AVMediaSelectionOption. Otherwise nil.

4.55.12 options as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A collection of mutually exclusive media selection options.

4.55.13 playableMediaSelectionOptionsFromArray(mediaSelectionOptions() as AVMediaSelectionOptionMBS) as AVMediaSelectionOptionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Filters an array of AVMediaSelectionOptions according to whether they are playable.
Notes: mediaSelectionOptions: An array of AVMediaSelectionOption to be filtered according to whether they are playable.

Returns an instance of NSArray containing the media selection options of the specified NSArray that are playable.

4.55.14 Properties

4.55.15 allowsEmptySelection as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether it's possible to present none of the options in the group when an associated AVPlayerItem is played.
Notes: If allowsEmptySelection is true, all of the available media options in the group can be deselected by passing nil as the specified AVMediaSelectionOption to AVPlayerItem.selectMediaOption.
(Read only property)

4.55.16 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.56 class AVMediaSelectionOptionMBS

4.56.1 class AVMediaSelectionOptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMediaSelectionOption represents a specific option for the presentation of media within a group of options.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.56.2 Methods

4.56.3 associatedMediaSelectionOptionInMediaSelectionGroup(mediaSelectionGroup as AVMediaSelectionGroupMBS) as AVMediaSelectionOptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: If a media selection option in another group is associated with the specified option, returns a reference to the associated option.
Notes: mediaSelectionGroup: A media selection group in which an associated option is to be sought.

Returns an instance of AVMediaSelectionOption.

Audible media selection options often have associated legible media selection options; in particular, audible options are typically associated with forced-only subtitle options with the same locale. See AVMediaCharacteristicContainsOnlyForcedSubtitles in AVMediaFormat.h for a discussion of forced-only subtitles.

4.56.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.56.5 availableMetadataFormats as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Provides an NSArray of NSStrings, each representing a metadata format that contains metadata associated with the option (e.g. ID3, iTunes metadata, etc.).

4.56.6 commonMetadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Provides an array of AVMetadataItems for each common metadata key for which a value is available.

4.56.7 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private construtor.

4.56.8 copy as AVMediaSelectionOptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.56.9 displayNameWithLocale(locale as NSLocaleMBS) as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Provides an string suitable for display.
Notes: Available on Mac OS X 10.9 or newer.
Localize manufactured portions of the string using the specificed locale.
May use this option's common metadata, media characteristics and locale properties in addition to the provided locale to formulate an NSString intended for display.

4.56.10 hasMediaCharacteristic(mediaCharacteristic as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Reports whether the media selection option includes media with the specified media characteristic.
Notes: The media characteristic of interest, e.g. AVMediaCharacteristicVisual, AVMediaCharacteristicAudible, AVMediaCharacteristicLegible, etc.

Returns ture if the media selection option includes media with the specified characteristic, otherwise false.

4.56.11 mediaSubTypes as Integer()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The mediaSubTypes of the media data associated with the option.
Notes: An Array of numbers carrying four character codes as defined in CoreAudioTypes.h for audio media and in CMFormatDescription.h for video media. Also see CMFormatDescriptionGetMediaSubType in CMFormatDescription.h for more information about media subtypes.

Note that if no information is available about the encoding of the media presented when a media option is selected, the value of mediaSubTypes will be an empty array. This can occur, for example, with streaming media. In these cases the value of mediaSubTypes should simply not be used as a criteria for selection.

4.56.12 metadataForFormat(Format as string) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Provides an Array of AVMetadataItems, one for each metadata item in the container of the specified format.
Notes: format: The metadata format for which items are requested.
An Array containing AVMetadataItems; may be nil if there is no metadata of the specified format.

4.56.13 propertyList as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a serializable property list that can be used to obtain an instance of AVMediaSelectionOption representing the same option as the receiver via AVMediaSelectionGroup.mediaSelectionOptionWithPropertyList.
Notes: Returns a serializable property list that's sufficient to identify the option within its group. For serialization utilities, see NSPropertyList.h.

4.56.14 Properties

4.56.15 displayName as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Provides an String suitable for display using the current system locale.
Notes: Available on Mac OS X 10.9 or newer.
May use this option's common metadata, media characteristics and locale properties in addition to the current system locale to formulate an string intended for display. Equivalent to displayNameWithLocale(NSLocaleMBS.currentLocale).
(Read only property)

4.56.16 extendedLanguageTag as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates the RFC 4646 language tag associated with the option. May be nil.
Notes: Available on Mac OS X 10.9 or newer.
(Read only property)

4.56.17 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.56.18 isPlayable as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether a media selection option is playable.
Notes: If the media data associated with the option cannot be decoded or otherwise rendered, playable is false.
(Read only property)

4.56.19 locale as NSLocaleMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the locale for which the media option was authored.
Notes: (Read only property)

4.56.20 mediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The media type of the media data, e.g. AVMediaTypeAudio, AVMediaTypeSubtitle, etc.
Notes: (Read only property)

4.57 class AVMetadataItemFilterMBS

4.57.1 class AVMetadataItemFilterMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: AVMetadataItemFilter is a tool used to filter AVMetadataItems.
Notes: Instances of AVMetadataItemFilter are used to filter AVMetadataItems. They are opaque, unmodifiable objects, created via AVMetadataItemFilter class methods.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr1

4.57.2 Methods

4.57.3 Add(keySpace as String, Key as String)

Plugin Version: 15.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds a new key to the whitelist.
Example:

```
dim e as AVAssetExportSessionMBS // your export session

// make new filter
dim n as new AVMetadataItemFilterMBS

// add new keys
n.Add(AVFoundationMBS.AVMetadataKeySpaceCommon, AVFoundationMBS.AVMetadataCommonKeyAuthor)
n.Add(AVFoundationMBS.AVMetadataKeySpaceCommon, AVFoundationMBS.AVMetadataCommonKeyTitle)
n.Add(AVFoundationMBS.AVMetadataKeySpaceCommon, AVFoundationMBS.AVMetadataCommonKeySubject)

// now use the filter
e.metadataItemFilter = n
```

4.57.4 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 and newer.

4.57.5 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor for a new empty filter.

4.57.6 metadataItemFilterForSharing as AVMetadataItemFilterMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Provides an instance of an AVMetadataItemFilter useful for sharing assets.
Notes: Removes many user-identifying metadata items, such as location information, leaving only playback-, copyright- and commercial-related metadata (such as the purchaser's Apple ID), along with metadata either derivable from the media itself or necessary for its proper behavior.

4.57.7 Properties

4.57.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.57.9 whitelist as Dictionary

Plugin Version: 15.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries white list for this filter.
Notes: This is dictionary. Keys are key spaces and values are arrays of keys.
(Read only property)

4.58 class AVMetadataItemMBS

4.58.1 class AVMetadataItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVMetadataItem object represents an item of metadata associated with an audiovisual asset or with one of its tracks.
Example:

```
// load file and look for cover art

Dim file As FolderItem = SpecialFolder.Desktop.Child("test.m4a")
Dim a As AVURLAssetMBS = AVURLAssetMBS.URLAssetWithFile(file)

Dim iTunesMetadataKeyCoverArtString As String = AVFoundationMBS.AVMetadataiTunesMetadataKeyCoverArt
Dim iTunesMetadataKeyCoverArtNumber As Integer = OSTypeFromStringMBS(iTunesMetadataKeyCoverArtString)

Dim items() As AVMetadataItemMBS = a.metadata

For Each item As AVMetadataItemMBS In items

Dim key As Variant = item.key
Dim Val As Variant = item.value

If key.IntegerValue = iTunesMetadataKeyCoverArtNumber Or key.StringValue = iTunesMetadataKeyCoverArtString Then

Dim v As MemoryBlock = Val
window1.Backdrop = picture.FromData(v)
End If

Next
```

Notes: To create metadata items for your own assets, you use the mutable subclass, AVMutableMetadataItem.

Metadata items have keys that accord with the specification of the container format from which they're drawn. Full details of the metadata formats, metadata keys, and metadata key spaces supported by AV Foundation are available among the defines in AVMetadataFormat.h.

You can load values of a metadata item "lazily" using the methods from the AVAsynchronousKeyValueLoading protocol (see "Asynchronous Loading"). The AVAsset class and other classes in turn provide their metadata lazily so that you can obtain objects from those arrays without incurring overhead for items you don't ultimately inspect.

You can filter arrays of metadata items using the methods of this class. For example, you can filter by key and key space, by locale, and by preferred language.

Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr1
- MBS Xojo / Real Studio Plugins, version 14.0pr6

Xojo Developer Magazine

- 12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.58.2 Methods

4.58.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.58.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.58.5 copy as AVMetadataItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.58.6 loadValuesAsynchronouslyForKeys(keys() as string, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Tells the asset to load the values of any of the specified keys that are not already loaded.
Notes: keys: An array containing the required keys.

Calls AVFoundationMBS.MetadataItemLoadValuesAsynchronouslyForKeysFinished on completion.
The completion event will be invoked exactly once per invocation of this method:

- Synchronously if an I/O error or other format-related error occurs immediately.
- Asynchronously at a subsequent time if a loading error occurs at a later stage of processing, or if cancelLoading is invoked on an AVAsset instance.

The completion states of the keys you specify in keys are not necessarily the same—some may be loaded, and others may have failed. You must check the status of each key individually.

If you want to receive error reporting for loading that's still pending, you can call this method at any time—even after an asset has begun to load data for operations in progress or already completed. If a fatal error has already occurred, the completion event is invoked synchronously.

4.58.7 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, key as Variant, keySpace as string) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns from a given array an array of metadata items that match a specified key or key space.
Notes: array: The array of AVMetadataItem objects to be filtered.
key: The key that must be matched for a metadata item to be included in the output array. The key is compared to the keys in the metadata in the array using isEqual. If you do not want to filter by key, pass nil.
keySpace: The key space that must be matched for a metadata item to be included in the output array. The key space is compared to the key spaces in the metadata in the array using isEqualToString:. If you do not want to filter by key space, pass nil.

Returns an array of the metadata items from array that match key or keySpace.

If the value in the keySpace parameter is AVMetadataKeySpaceCommon, the value of the metadata item's commonKey property must match the value in the key parameter for the item to be returned. If the key parameter is nil, then all metadata items in the common key space are returned.

If the value in the keySpace parameter is nil or is not the constant for the common key space, the value of the metadata item's key property must match the value in the key parameter for the item to be returned. If the key parameter itself is nil, then all metadata items in the specified key space are returned.
See also:

- 4.58.8 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, locale as NSLocaleMBS) as AVMetadataItemMBS() 494
- 4.58.9 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, metadataItemFilter as AVMetadataItemFilterMBS) as AVMetadataItemMBS() 494
- 4.58.10 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, preferredLanguages() as string) as AVMetadataItemMBS() 495

4.58.8 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, locale as NSLocaleMBS) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns from a given array an array of metadata items that match a specified locale.
Notes: array: The array of AVMetadataItem objects to be filtered.
locale: The locale that must be matched for a metadata item to be included in the output array.

Returns an array containing the AVMetadataItem objects from the array parameter that match the specified locale.
See also:

- 4.58.7 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, key as Variant, keySpace as string) as AVMetadataItemMBS() 493
- 4.58.9 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, metadataItemFilter as AVMetadataItemFilterMBS) as AVMetadataItemMBS() 494
- 4.58.10 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, preferredLanguages() as string) as AVMetadataItemMBS() 495

4.58.9 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, metadataItemFilter as AVMetadataItemFilterMBS) as AVMetadataItemMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Filters an array of metadata items using the metadata item filter.
Notes: metadataItems: An array of AVMetadataItems to be filtered.
metadataItemFilter: The AVMetadataItemFilter object for filtering the metadataItems.

Returns an instance of NSArray containing the metadata items in metadataItems that have not been removed by metadataItemFilter.
Available in OS X v10.9 and later.
See also:

- 4.58.7 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, key as Variant, keySpace as string) as AVMetadataItemMBS() 493

- 4.58.8 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, locale as NSLocaleMBS) as AVMetadataItemMBS() 494
- 4.58.10 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, preferredLanguages() as string) as AVMetadataItemMBS() 495

4.58.10 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, preferredLanguages() as string) as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the subset of metadata items whose locales match one of the specified language identifiers.
Notes: metadataItems: An array of AVMetadataItem objects to be filtered and sorted.
preferredLanguages: An array of strings, each of which contains a canonicalized IETF BCP 47 language identifier. The order of the identifiers in the array reflects the preferred language order, with the most preferred language being first in the array. Typically, you pass the user's preferred languages by retrieving this array from the preferredLanguages class method of NSLocale.

Returns an array containing the AVMetadataItem objects from the metadataItems parameter that match one of the specified languages.
See also:

- 4.58.7 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, key as Variant, keySpace as string) as AVMetadataItemMBS() 493
- 4.58.8 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, locale as NSLocaleMBS) as AVMetadataItemMBS() 494
- 4.58.9 metadataItemsFromArray(metadataItems() as AVMetadataItemMBS, metadataItemFilter as AVMetadataItemFilterMBS) as AVMetadataItemMBS() 494

4.58.11 mutableCopy as AVMutableMetadataItemMBS

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Creates a mutable copy.

4.58.12 statusOfValueForKey(key as string, byref error as NSErrorMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Reports whether the value for a given key is immediately available without blocking. (required)
Notes: key: The key whose status you want.

error: If the status of the value for the key is AVKeyValueStatusFailed, upon return contains an NSError object that describes the failure that occurred.

Returns the current loading status of the value for key. For possible values, see ”Protocol Methods.”

You use this method to determine the availability of the value for a key. This method does not cause an asset to load the value of a key that’s not yet available. To request values for keys that may not already be loaded without blocking, use loadValuesAsynchronouslyForKeys and wait for invocation of the completion handler to be informed of availability.

4.58.13 Properties

4.58.14 commonKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The common key of the metadata item. (read-only)
Notes: This property contains the key that most closely corresponds to the key in the key property but that belongs to the common key space (AVMetadataKeySpaceCommon) as opposed to a format-specific key space. You can use this key to locate metadata items irrespective of the underlying media format.

If the value of the keySpace property is AVMetadataKeySpaceCommon, this property contains the same key as the key property.
(Read only property)

4.58.15 dataValue as Memoryblock

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Provides the raw bytes of the value of the metadata item. (read-only)
Notes: (Read only property)

4.58.16 dateTimeValue as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Provides the raw bytes of the value of the metadata item. (read-only)
Notes: (Read only property)

4.58.17 dateValue as date

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The value of the metadata item formatted as a date. (read-only)
Notes: The value of this property is nil if the metadata value cannot be represented as a date.
(Read only property)

4.58.18 duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The duration of the metadata item. (read-only)
Notes: (Read only property)

4.58.19 extraAttributes as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The additional attributes supplied by the metadata item. (read-only)
Notes: (Read only property)

4.58.20 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.58.21 key as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key of the metadata item. (read-only)
Notes: The key property contains the true key used to identify the contents of the metadata item. This value is specific to the key space of the metadata item.
(Read only property)

4.58.22 keySpace as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The key space of metadata item's key. (read-only)
Notes: The key space specified by this property is typically the default key space for the metadata container in which the metadata item is stored.

AV Foundation uses key spaces to group related sets of keys. For example, the framework defines different key spaces for common keys, iTunes keys, ID3 keys, and QuickTime keys. Key spaces aid in filtering arrays of metadata items.
(Read only property)

4.58.23 locale as NSLocaleMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The locale of the metadata item. (read-only)
Notes: The locale may be nil if no locale information is available for the metadata item.
(Read only property)

4.58.24 numberValue as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The value of the metadata item formatted as a number. (read-only)
Notes: The value of this property is nil if the metadata value cannot be represented as a number.
(Read only property)

4.58.25 stringValue as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The value of the metadata item formatted as a string. (read-only)
Notes: The value of this property is nil if the metadata value cannot be represented as a string.
(Read only property)

4.58.26 time as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The timestamp of the metadata item. (read-only)
Notes: (Read only property)

4.58.27 value as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The value of the metadata item. (read-only)
Notes: (Read only property)

4.58.28 Constants

Load Status Constants

Constant	Value	Description
AVKeyValueStatusCancelled	4	Indicates that the attempt to load the property was cancelled.
AVKeyValueStatusFailed	3	Indicates that the attempt to load the property failed.
AVKeyValueStatusLoaded	2	Indicates that the property is ready for use.
AVKeyValueStatusLoading	1	Indicates that the property is not fully loaded.
AVKeyValueStatusUnknown	0	Indicates that the property status is unknown.

4.59 class AVMetadataObjectMBS

4.59.1 class AVMetadataObjectMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The class for a metadata object.
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 21.2

4.59.2 Methods

4.59.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.

4.59.4 AVMetadataObjectTypeAztecCode as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Aztec codes

4.59.5 AVMetadataObjectTypeCatBody as string

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

Function: One of the types of metadata.
Notes: The type identifier for a metadata object representing a cat body in a picture.

4.59.6 AVMetadataObjectTypeCode128Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Code 128

4.59.7 AVMetadataObjectTypeCode39Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Code 39

4.59.8 AVMetadataObjectTypeCode39Mod43Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Code 39 mod 43

4.59.9 AVMetadataObjectTypeCode93Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Code 93

4.59.10 AVMetadataObjectTypeDataMatrixCode as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from DataMatrix

4.59.11 AVMetadataObjectTypeDogBody as string

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

Function: One of the types of metadata.
Notes: The type identifier for a metadata object representing a dog body in a picture.

4.59.12 AVMetadataObjectTypeEAN13Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from EAN-13 (including UPC-A) codes

4.59.13 AVMetadataObjectTypeEAN8Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from EAN-8 codes

4.59.14 AVMetadataObjectTypeFace as string

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

Function: One of the types of metadata.
Notes: AVMetadataFaceObject instances representing the features of a detected face

4.59.15 AVMetadataObjectTypeHumanBody as string

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

Function: One of the types of metadata.
Notes: The type identifier for a metadata object representing a human body in a picture.

4.59.16 AVMetadataObjectTypeInterleaved2of5Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from Interleaved 2 of 5 codes

4.59.17 AVMetadataObjectTypeITF14Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from ITF14 codes

4.59.18 AVMetadataObjectTypePDF417Code as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from PDF417 codes

4.59.19 AVMetadataObjectTypeQRCode as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from QR codes

4.59.20 AVMetadataObjectTypeSalientObject as string

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

Function: One of the types of metadata.
Notes: The type identifier for a metadata object representing a single salient area in a picture.

4.59.21 AVMetadataObjectTypeUPCECode as string

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

Function: One of the types of metadata.
Notes: AVMetadataMachineReadableCodeObject instances generated from UPC-E codes

4.59.22 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.59.23 corners as Dictionary()

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

Function: The points defining the (x, y) locations of the corners.
Notes: The value of this property is an array of Dictionary objects, each of which has been created from

a CGPoint struct using the CGPointCreateDictionaryRepresentation function, representing the coordinates of the corners of the object with respect to the image in which it resides.
If the metadata originates from video, the points may be expressed as scalar values from 0 to 1.
The points in the corners differ from the bounds rectangle in that bounds is axis aligned to orientation of the captured image, and the values of the corners reside within the bounds rectangle.
The points are arranged in counterclockwise order (clockwise if the code or image is mirrored), starting with the top left of the code in its canonical orientation.

For barcodes.

4.59.24 Properties

4.59.25 barcodeDescriptor as Variant

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

Function: A barcode description for use in Core Image.
Notes: (Read only property)

4.59.26 bodyID as Integer

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

Function: An integer value that defines the unique identifier of an object in a picture.
Notes: For bodys including human, dog or cat.
(Read only property)

4.59.27 Bounds as CGRectMBS

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

Function: The bounding rectangle associated with the metadata.
Notes: The bounding rectangle is specified relative to the picture or video of the corresponding media. The rectangle,Äôs origin is always specified in the top-left corner, and the x and y axis extend down and to the right.
If the metadata has no bounding rectangle, the value of this property should be CGRectZero.
For video content, the bounding rectangle may be expressed using scalar values in the range 0.0 to 1.0. Scalar values remain meaningful even when the original video has been scaled down.

Available on macOS 10.10 or later.
(Read only property)

4.59.28 Duration as CMTimeMBS

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

Function: The duration of the media associated with this metadata object.
Notes: For metadata originating from a sample buffer (CMSampleBuffer), the duration reflects the duration of the sample buffer. If there is no valid duration value associated with the metadata, this property should contain invalid.

Available on macOS 10.10 or later.
(Read only property)

4.59.29 faceID as Integer

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

Function: The unique ID for this face metadata object.
Notes: Each time a face enters the picture, it is assigned a new unique identifier, which you can use to reference the face in your code. Face IDs are not reused, and the same face leaving and entering the picture again is assigned a new identifier.

For face objects only.
(Read only property)

4.59.30 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.59.31 hasRollAngle as Boolean

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

Function: A Boolean value indicating whether there is a valid roll angle associated with the face.
Notes: If the value of this property is false, the value in the rollAngle property is invalid and must not be accessed.

For face objects only.
(Read only property)

4.59.32 hasYawAngle as Boolean

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

Function: A Boolean value indicating whether there is a valid yaw angle associated with the face.
Notes: If the value of this property is false, the value in the yawAngle property is invalid and must not be accessed.

For face objects only.
(Read only property)

4.59.33 objectID as Integer

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

Function: An integer value that defines the unique identifier of an object in a picture.
Notes: When an object enters the picture, it‚Äôs assigned a new, unique identifier.

For salient objects only.
(Read only property)

4.59.34 rollAngle as Double

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

Function: The roll angle of the face specified in degrees.
Notes: The roll angle represents the side-to-side tilt of the face relative to the metadata‚Äôs bounding rectangle. A value of 0.0 yields a face that is level relative to the picture, whereas a value of 90 yields a face that is perpendicular relative to the picture.

You must check the value of the hasRollAngle property before accessing this property. If the value in the hasRollAngle property is false, reading the value in this property raises an exception.

For face objects only.
(Read only property)

4.59.35 stringValue as String

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

Function: Returns the error-corrected data decoded into a human-readable string.
Notes: For barcodes.

The value of this property is a string created by decoding the binary payload according to the format of the machine-readable code, or "" if a string representation cannot be created.
(Read only property)

4.59.36 Time as CMTimeMBS

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

Function: The media time value associated with the metadata object.
Notes: For captured media, this property represents the time when the metadata was captured. For metadata originating from a sample buffer (CMSampleBuffer), the time is the sample buffer‚Äôs presentation time. If there is no valid time value associated with the metadata, this property should contain invalid.

Available on macOS 10.10 or later.
(Read only property)

4.59.37 Type as String

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

Function: The type of the metadata.
Notes: You use this property value to distinguish between different types of metadata. The possible values for this property are defined by subclasses, which can define constants for the type of metadata they represent.
(Read only property)

4.59.38 yawAngle as Double

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

Function: The yaw angle of the face specified in degrees.
Notes: The yaw angle represents the rotation of the face around the vertical axis. A value of 0.0 yields a face that is looking directly at the camera, whereas a yaw angle of 90 degrees yields a face whose eye line is perpendicular to that of the camera.

You must check the value of the hasYawAngle property before accessing this property. If the value in the hasYawAngle property is NO, reading the value in this property raises an exception.

For face objects only.
(Read only property)

4.60 class AVMutableAudioMixInputParametersMBS

4.60.1 class AVMutableAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVMutableAudioMixInputParameters object represents the parameters that should be applied to an audio track when it is added to a mix.
Notes: Subclass of the AVAudioMixInputParametersMBS class.

4.60.2 Methods

4.60.3 audioMixInputParameters as AVMutableAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates and returns a mutable input parameters object with no volume ramps and trackID initialized to kCMPersistentTrackID_Invalid.

4.60.4 audioMixInputParametersWithTrack(track as AVAssetTrackMBS) as AVMutableAudioMixInputParametersMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates and returns a mutable input parameters object for a given track.
Notes: track: The track to associate with the input parameters object.

Returns a mutable input parameters object with no volume ramps and trackID set to track's ID.

4.60.5 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates and returns a mutable input parameters object with no volume ramps and trackID initialized to kCMPersistentTrackID_Invalid.
See also:

- 4.60.6 Constructor(other as AVAudioMixInputParametersMBS) 510
- 4.60.7 Constructor(track as AVAssetTrackMBS) 510

4.60.6 Constructor(other as AVAudioMixInputParametersMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.60.5 Constructor 509
- 4.60.7 Constructor(track as AVAssetTrackMBS) 510

4.60.7 Constructor(track as AVAssetTrackMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates and returns a mutable input parameters object for a given track.
Notes: track: The track to associate with the input parameters object.
Initializes input parameters object with no volume ramps and trackID set to track's ID.
See also:

- 4.60.5 Constructor 509
- 4.60.6 Constructor(other as AVAudioMixInputParametersMBS) 510

4.60.8 setVolume(volume as Double, atTime as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the value of the audio volume starting at the specified time.
Notes: volume: The volume.
time: The start time at which to set the volume.

This method adds a volume ramp starting at time. This volume setting remains in effect until the end of the track unless you set a different volume level to start at a later time.

4.60.9 setVolumeRamp(startVolume as Double, endVolume as Double, timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets a volume ramp to apply during a specified time range.
Notes: startVolume: The starting volume.
endVolume: The end volume.
timeRange: The time range over which to apply the ramp.

4.60.10 Properties

4.60.11 trackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The ID of the audio track to which the parameters should be applied.
Notes: (Read and Write computed property)

4.61 class AVMutableAudioMixMBS

4.61.1 class AVMutableAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVMutableAudioMix object manages the input parameters for mixing audio tracks.
Notes: It allows custom audio processing to be performed on audio tracks during playback or other operations.
Subclass of the AVAudioMixMBS class.

4.61.2 Methods

4.61.3 audioMix as AVMutableAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new mutable audio mix.

4.61.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.61.5 Constructor(other as AVAudioMixMBS) 512

4.61.5 Constructor(other as AVAudioMixMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.61.4 Constructor 512

4.61.6 setInputParameters(items() as AVAudioMixInputParametersMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the parameters for inputs to the mix.
Notes: The array contains instances of AVAudioMixInputParameters. Note that an instance of AVAudioMixInputParameters is not required for each audio track that contributes to the mix; audio for those without associated AVAudioMixInputParameters will be included in the mix, processed according to default behavior.

4.62 class AVMutableCompositionMBS

4.62.1 class AVMutableCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMutableComposition is a mutable subclass of AVComposition you use when you want to create a new composition from existing assets.
Notes: You can add and remove tracks, and you can add, remove, and scale time ranges.
You can make an immutable snapshot of a mutable composition for playback or inspection with the copy method.
Subclass of the AVCompositionMBS class.
Blog Entries

- MBS Xojo Plugins, version 18.2pr5
- Automated video editing with Xojo

4.62.2 Methods

4.62.3 addMutableTrackWithMediaType(mediaType as string, preferredTrackID as Integer) as AVMutableCompositionTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds an empty track to the receiver.
Notes: mediaType: The media type of the new track.
preferredTrackID: The preferred track ID for the new track. If you do not need to specify a preferred track ID, pass kCMPersistentTrackID_Invalid. The preferred track ID will be used for the new track provided that it is not currently in use and has not previously been used. If the preferred track ID you specify is not available, or if you pass in kCMPersistentTrackID_Invalid, a unique track ID is generated.

Returns an instance of AVMutableCompositionTrack representing the new track.
You can get the actual trackID of the new track through its "trackID" key.

4.62.4 append(asset as AVAssetMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 18.2, Platform: macOS, Targets: All.

Function: Appends all the tracks to the receiver.
Notes: asset: An asset that contains the tracks to be inserted.
Error: If the insertion was not successful, on return contains an NSError object that describes the problem.

Return true if the insertion was successful, otherwise false.

This method may add new tracks to ensure that all tracks of the asset are represented in the inserted time range.
Existing content at the specified start time is pushed out by the duration of the time range.
Media data for the inserted time range is presented at its natural duration; you can scale it to a different duration using scaleTimeRange.

4.62.5 appendTimeRange(timeRange as CMTimeRangeMBS, asset as AVAssetMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 18.2, Platform: macOS, Targets: All.

Function: Appends all the tracks within a given time range of a specified asset into the receiver.
Notes: timeRange: The time range of the asset to be inserted.
asset: An asset that contains the tracks to be inserted.
Error: If the insertion was not successful, on return contains an NSError object that describes the problem.

Return true if the insertion was successful, otherwise false.

This method may add new tracks to ensure that all tracks of the asset are represented in the inserted time range.
Existing content at the specified start time is pushed out by the duration of the time range.
Media data for the inserted time range is presented at its natural duration; you can scale it to a different duration using scaleTimeRange.

4.62.6 composition as AVMutableCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new, empty, mutable composition.

4.62.7 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.62.8 Constructor(other as AVCompositionMBS) 516

4.62.8 Constructor(other as AVCompositionMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.62.7 Constructor 515

4.62.9 insert(asset as AVAssetMBS, startTime as CMTimeMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 18.2, Platform: macOS, Targets: All.

Function: Inserts all the tracks of a specified asset into the receiver.
Notes: asset: An asset that contains the tracks to be inserted.
startTime: The time at which the inserted tracks should be presented by the receiver.
Error: If the insertion was not successful, on return contains an NSError object that describes the problem.

Return true if the insertion was successful, otherwise false.

This method may add new tracks to ensure that all tracks of the asset are represented in the inserted time range.
Existing content at the specified start time is pushed out by the duration of the time range.
Media data for the inserted time range is presented at its natural duration; you can scale it to a different duration using scaleTimeRange.

4.62.10 insertEmptyTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds or extends an empty timeRange within all tracks of the composition.
Notes: timeRange: The empty time range to insert.

If you insert an empty time range into the composition, any media that was presented during that interval prior to the insertion will be presented instead immediately afterward. You can use this method to reserve an interval in which you want a subsequently created track to present its media.

4.62.11 insertTimeRange(timeRange as CMTimeRangeMBS, asset as AVAssetMBS, startTime as CMTimeMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Inserts all the tracks within a given time range of a specified asset into the receiver.
Notes: timeRange: The time range of the asset to be inserted.
asset: An asset that contains the tracks to be inserted.
startTime: The time at which the inserted tracks should be presented by the receiver.
Error: If the insertion was not successful, on return contains an NSError object that describes the problem.

Return true if the insertion was successful, otherwise false.

This method may add new tracks to ensure that all tracks of the asset are represented in the inserted time range.
Existing content at the specified start time is pushed out by the duration of the time range.
Media data for the inserted time range is presented at its natural duration; you can scale it to a different duration using scaleTimeRange.

4.62.12 MutableCompositionTracks as AVMutableCompositionTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVMutableCompositionTrack objects contained by the composition. (read-only)

4.62.13 mutableTrackCompatibleWithTrack(track as AVAssetTrackMBS) as AVMutableCompositionTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a track in the receiver into which any time range of a given asset track can be inserted.
Notes: track: An AVAssetTrack from which a time range may be inserted.

Returns a mutable track in the receiver into which any time range of track can be inserted. If no such track is available, the returns nil.

For best performance, you should keep the number of tracks of a composition should be kept to a minimum, corresponding to the number for which media data must be presented in parallel. If you want to present media data of the same type serially, even from multiple assets, you should use a single track of that media type. You use this method to identify a suitable existing target track for an insertion.

If there is no compatible track available, you can create a new track of the same media type as track using addMutableTrackWithMediaType.

This method is similar to compatibleTrackForCompositionTrack (AVAsset).

4.62.14 removeTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a specified timeRange from all tracks of the composition.
Notes: timeRange: The time range to be removed.

After removing, existing content after the time range will be pulled in.
Removal of a time range does not cause any existing tracks to be removed from the composition, even if removing timeRange results in an empty track. Instead, it removes or truncates track segments that intersect with the time range.

4.62.15 removeTrack(track as AVCompositionTrackMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a specified track from the receiver.
Notes: track: The track to remove.

When it is removed track's @"composition" key is set to nil. The values of its other keys remain intact, for arbitrary use.

4.62.16 scaleTimeRange(timeRange as CMTimeRangeMBS, duration as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Changes the duration of all tracks in a given time range.
Notes: timeRange: The time range of the composition to be scaled.
duration: The new duration of timeRange.

Each track segment affected by the scaling operation will be presented at a rate equal to source.duration / target.duration of its resulting time mapping.

4.62.17 Properties

4.62.18 naturalSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The encoded or authored size of the visual portion of the asset.
Notes: If this value is not set, the default behavior is as defined by AVAsset; set the value to CGSizeZero to revert to the default behavior.
Available in OS X v10.7 and later.
Deprecated in OS X v10.8.
(Read and Write computed property)

4.63 class AVMutableCompositionTrackMBS

4.63.1 class AVMutableCompositionTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMutableCompositionTrack is a mutable subclass of AVCompositionTrack that lets you for insert, remove, and scale track segments without affecting their low-level representation (that is, the operations you perform are non-destructive on the original).
Notes: AVCompositionTrack defines constraints for the temporal alignment of the track segments. If you set the array of track segments in a mutable composition (see trackSegments), you can test whether the segments meet the constraints using validateTrackSegments.
Subclass of the AVCompositionTrackMBS class.

4.63.2 Methods

4.63.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 4.63.4 Constructor(other as AVCompositionTrackMBS) 520

4.63.4 Constructor(other as AVCompositionTrackMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.63.3 Constructor 520

4.63.5 insertEmptyTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds or extends an empty time range within the receiver.
Notes: timeRange: The empty time range to be inserted.

If you insert an empty time range into the track, any media that was presented during that interval prior to the insertion will be presented instead immediately afterward.

The nature of the data inserted depends upon the media type of the track. For example, an empty time range in a sound track presents silence.

4.63.6 insertTimeRange(timeRange as CMTimeRangeMBS, AssetTrack as AVAssetTrackMBS, startTime as CMTimeMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Inserts a time range of a source track.
Notes: timeRange: The time range of the track to be inserted.
track: The source track to be inserted.
startTime: The time at which track is to be presented by the composition track.
error: If track is not inserted successfully, contains an NSError object that describes the problem.

Returns true if track was inserted successfully, otherwise false.

By default, the inserted track's time range is presented at its natural duration and rate. You can scale it to a different duration (so that it is presented at a different rate) using scaleTimeRange:toDuration:.

Insertion might fail if, for example, the asset that you try to insert is restricted by copy-protection.

4.63.7 insertTimeRanges(timeRanges() as CMTimeRangeMBS, tracks() as AVAssetTrackMBS, startTime as CMTimeMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Inserts time ranges of source tracks.
Notes: Requires Mac OS X 10.8.

Inserts the timeRanges of multiple source tracks into a track of a composition.
timeRanges: Specifies the timeRanges to be inserted.
tracks: Specifies the source tracks to be inserted. Only AVAssetTracks of AVURLAssets are supported.
startTime: Specifies the time at which the inserted tracks are to be presented by the composition track. You may pass kCMTimeInvalid for startTime to indicate that the timeRanges should be appended to the end of the track.
error: Describes failures that may be reported to the user, e.g. the asset that was selected for insertion in the composition is restricted by copy-protection.
Returns a boolean value indicating the success of the insertion.

This method is equivalent to (but more efficient than) calling insertTimeRange for each timeRange/track pair.
If this method returns an error, none of the time ranges will be inserted into the composition track.
To specify an empty time range, pass NSNull for the track and a time range of starting at kCMTimeInvalid with a duration of the desired empty edit.

4.63.8 removeTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a specified time range from the receiver.
Notes: timeRange: The time range to be removed.

Removing a time range does not cause the track to be removed from the composition. Instead it removes or truncates track segments that intersect with the time range.

4.63.9 scaleTimeRange(timeRange as CMTimeRangeMBS, duration as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Changes the duration of a time range in the receiver.
Notes: timeRange: The time range of the track to be scaled.
duration: The new duration of timeRange.

Each track segment affected by the scaling operation will be presented at a rate equal to source.duration / target.duration of its resulting timeMapping.

4.63.10 setCompositionTrackSegments(segments() as AVCompositionTrackSegmentMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the composition track's array of track segments.
Notes: The timeMapping.target.start of the first track segment must be kCMTimeZero, and the timeMapping.target.start of each subsequent track segment must equal CMTimeRangeGetEnd(<#previousTrackSegment#>.timeMapping.target). You can use validateTrackSegments to ensure that an array of track segments conforms to this rule.

4.63.11 validateTrackSegments(trackSegments() as AVCompositionTrackSegmentMBS, byref error as NSErrorMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given array of track segments conform to the timing rules for a composition track.
Notes: trackSegments: An array of AVCompositionTrackSegment objects.
error: If validation fails, on return contains an NSError object that describes the problem.

Returns true if the track segments in trackSegments conform to the timing rules for a composition track, otherwise false.

You can use this method to ensure that an array of track segments is suitable for setting as the value of the trackSegments property. The timeMapping.target.start of the first track segment must be kCMTimeZero, and the timeMapping.target.start of each subsequent track segment must equal CMTimeRangeGetEnd(<#previousTrackSegment#>.timeMapping.target).

If you want to modify the existing trackSegments array, you can create a mutable copy of it, modify the mutable array, and then validate the mutable array using this method.

4.63.12 Properties

4.63.13 extendedLanguageTag as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The language tag associated with the track, as an RFC 4646 language tag.
Notes: If not set, the value is nil.
(Read and Write computed property)

4.63.14 languageCode as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The language associated with the track, as an ISO 639-2/T language code.
Notes: If not set, the value is nil.
(Read and Write computed property)

4.63.15 naturalTimeScale as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The timescale in which time values for the track can be operated upon without extraneous numerical conversion.
Notes: If not set, the value is the natural time scale of the first non-empty edit, or 600 if there are no non-empty edits.

Set the value to 0 to revert to the default behavior.
(Read and Write computed property)

4.63.16 preferredTransform as CGAffineTransformMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The preferred transformation of the visual media data for display purposes.
Notes: If not set, the value is CGAffineTransformIdentity.
(Read and Write computed property)

4.63.17 preferredVolume as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The preferred volume of the audible media data.
Notes: If not set, the value is 1.0.
(Read and Write computed property)

4.64 class AVMutableMetadataItemMBS

4.64.1 class AVMutableMetadataItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMutableMetadataItem is a mutable subclass of AVMetadataItem that lets you build collections of metadata to be written to asset files using AVAssetExportSession.
Notes: You can initialize a mutable metadata item from an existing AVMetadataItem object or with a one or more of the basic properties of a metadata item: a key, a key space, a locale, and a value.
Subclass of the AVMetadataItemMBS class.

4.64.2 Methods

4.64.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
Example:

```
dim e as AVAssetExportSessionMBS // your export session
dim asset as AVAssetMBS // your asset

// query metadata
dim a() as AVMetadataItemMBS = asset.metadata

// make new
dim m as new AVMutableMetadataItemMBS

// add common key with author
m.keySpace = AVFoundationMBS.AVMetadataKeySpaceCommon
m.key = AVFoundationMBS.AVMetadataCommonKeyAuthor
m.Value = "Hello World"

// append to array and use as metadata:
a.Append m
e.setMetadata a
```

See also:

- 4.64.4 Constructor(other as AVMetadataItemMBS) 526

4.64.4 Constructor(other as AVMetadataItemMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.64.3 Constructor 525

4.64.5 metadataItem as AVMutableMetadataItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new mutable metadata item.

4.64.6 Properties

4.64.7 duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the metadata item's duration.
Notes: (Read and Write computed property)

4.64.8 extraAttributes as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Provides a dictionary of the metadata item's additional attributes.
Notes: (Read and Write computed property)

4.64.9 key as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the metadata item's key.
Notes: (Read and Write computed property)

4.64.10 keySpace as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the key space of the metadata item's key.
Notes: This is typically the default key space for the metadata container in which the metadata item is stored.
(Read and Write computed property)

4.64.11 locale as NSLocaleMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the metadata item's locale.
Notes: The locale may be nil if no locale information is available for the item.
(Read and Write computed property)

4.64.12 time as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the metadata item's timestamp.
Notes: (Read and Write computed property)

4.64.13 value as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the metadata item's value.
Notes: (Read and Write computed property)

4.65 class AVMutableTimedMetadataGroupMBS

4.65.1 class AVMutableTimedMetadataGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVMutableTimedMetadataGroup object to represent a mutable collection of metadata items.
Notes: Subclass of the AVTimedMetadataGroupMBS class.

4.65.2 Methods

4.65.3 Constructor(items() as AVMetadataItemMBS, timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.65.4 Constructor(other as AVTimedMetadataGroupMBS) 528

4.65.4 Constructor(other as AVTimedMetadataGroupMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.65.3 Constructor(items() as AVMetadataItemMBS, timeRange as CMTimeRangeMBS) 528

4.65.5 items as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The metadata items in the group.

4.65.6 setItems(items() as AVMetadataItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the metadata items in the group.

4.65.7 Properties

4.65.8 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range of the metadata.
Notes: (Read and Write computed property)

4.66 class AVMutableVideoCompositionInstructionMBS

4.66.1 class AVMutableVideoCompositionInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVMutableVideoCompositionInstruction object represents an operation to be performed by a compositor.
Notes: An AVVideoComposition object maintains an array of instructions to perform its composition.
Subclass of the AVVideoCompositionInstructionMBS class.
Blog Entries

- Automated video editing with Xojo

4.66.2 Methods

4.66.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.66.4 Constructor(other as AVVideoCompositionInstructionMBS) 530

4.66.4 Constructor(other as AVVideoCompositionInstructionMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.66.3 Constructor 530

4.66.5 layerInstructions as AVVideoCompositionLayerInstructionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of instances of AVVideoCompositionLayerInstruction that specify how video frames from source tracks should be layered and composed.
Notes: Tracks are layered in the composition according to the top-to-bottom order of the layerInstructions array; the track with trackID of the first instruction in the array will be layered on top, with the track with the trackID of the second instruction immediately underneath, and so on.

If the property value is nil, the output is a fill of the background color.

4.66.6 setLayerInstructions(items() as AVVideoCompositionLayerInstructionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets an array of instances of AVVideoCompositionLayerInstruction that specify how video frames from source tracks should be layered and composed.
Notes: Tracks are layered in the composition according to the top-to-bottom order of the layerInstructions array; the track with trackID of the first instruction in the array will be layered on top, with the track with the trackID of the second instruction immediately underneath, and so on.

If the property value is nil, the output is a fill of the background color.

4.66.7 videoCompositionInstruction as AVMutableVideoCompositionInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new mutable video composition instruction.

4.66.8 Properties

4.66.9 backgroundColor as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The background color of the composition.
Notes: Value must be CGColorMBS.
Only solid BGRA colors are supported; patterns and other supported colors are ignored. If the rendered pixel buffer does not have alpha, the alpha value of the background color is ignored.

If the background color is nil, the video compositor uses a default background color of opaque black.
(Read and Write computed property)

4.66.10 enablePostProcessing as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether post processing is required for the video composition instruction.
Notes: If no post processing is required for the whole duration of the video composition instruction, set this property to false to make the composition process more efficient.

The value is true by default.
(Read and Write computed property)

4.66.11 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range during which the instruction is effective.
Notes: If the time range is invalid, the video compositor will ignore it.
(Read and Write computed property)

4.67 class AVMutableVideoCompositionLayerInstructionMBS

4.67.1 class AVMutableVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVMutableVideoCompositionLayerInstruction is a mutable subclass of AVVideoCompositionLayerInstruction that you use to modify the transform and opacity ramps to apply to a given track in an AV composition.
Notes: Subclass of the AVVideoCompositionLayerInstructionMBS class.
Blog Entries

- Automated video editing with Xojo
- MBS Xojo / Real Studio Plugins, version 15.0pr12

4.67.2 Methods

4.67.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.
See also:

- 4.67.4 Constructor(other as AVMutableVideoCompositionLayerInstructionMBS) 533

4.67.4 Constructor(other as AVMutableVideoCompositionLayerInstructionMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.67.3 Constructor 533

4.67.5 setCropRectangle(cropRectangle as CGRectMBS, time as CMTimeMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Sets a value of the crop rectangle at a time within the time range of the instruction.
Notes: cropRectangle: The crop rectangle to be applied at the specified time.
time: A time value within the timeRange of the composition instruction.

The origin of the crop rectangle is the top-left corner of the buffer clean aperture rectangle. The crop rectangle is defined in square pixel space, that is, without taking the pixel aspect ratio into account. Crop rectangles extending outside of the clean aperture, are cropped to the clean aperture.

Sets a fixed crop rectangle to apply from time until the next time at which a crop rectangle is set; this is the same as setting a flat ramp for that time range.

Before the first specified time for which a crop rectangle is set, the crop rectangle is held constant to CGRectInfinite and after the last time for which a crop rectangle is set, the crop rectangle is held constant at that last value.
Available in OS X v10.9 and later.

4.67.6 setCropRectangleRampFromStartCropRectangle(startCropRectangle as CGRectMBS, endCropRectangle as CGRectMBS, timeRange as CMTimeRangeMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Sets a crop rectangle ramp to apply during the specified time range.
Notes: startCropRectangle: The crop rectangle to be applied at the starting time of the timeRange.
endCropRectangle: The crop rectangle to be applied at the end time of the timeRange.
timeRange: The time range over which the value of the opacity is interpolated between startCropRectangle and endCropRectangle.

The origin of the crop rectangle is the top-left corner of the buffer clean aperture rectangle. The crop rectangle is defined in square pixel space, that is, without taking the pixel aspect ratio into account. Crop rectangles extending outside of the clean aperture, are cropped to the clean aperture.

During a crop rectangle ramp, the rectangle is interpolated between the values set at the ramp's start time and end time. When the starting or ending rectangle is empty, interpolations take into account the origin and size of the empty rectangle.

Before the first specified time for which a crop rectangle is set, the crop rectangle is held constant to CGRectInfinite and after the last time for which a crop rectangle is set, the crop rectangle is held constant at that last value.
Available in OS X v10.9 and later.

4.67.7 setOpacity(opacity as Double, time as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets a value of the opacity at a time within the time range of the instruction.
Notes: opacity: The opacity to be applied at time. The value must be between 0.0 and 1.0.
time: A time value within the time range of the composition instruction.

Sets a fixed opacity to apply from the specified time until the next time at which an opacity is set; this is the same as setting a flat ramp for that time range. Before the first time for which an opacity is set, the opacity is held constant at 1.0; after the last specified time, the opacity is held constant at the last value.
See also:

- 4.67.8 setOpacity(startOpacity as Double, endOpacity as Double, timeRange as CMTimeRangeMBS) 535

4.67.8 setOpacity(startOpacity as Double, endOpacity as Double, timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets an opacity ramp to apply during a specified time range.
Notes: startOpacity: The opacity to be applied at the start time of timeRange. The value must be between 0.0 and 1.0.
endOpacity: The opacity to be applied at the end time of timeRange. The value must be between 0.0 and 1.0.
timeRange: The time range over which the value of the opacity will be interpolated between startOpacity and endOpacity.

During an opacity ramp, opacity is computed using a linear interpolation. Before the first time for which an opacity is set, the opacity is held constant at 1.0; after the last specified time, the opacity is held constant at the last value.
See also:

- 4.67.7 setOpacity(opacity as Double, time as CMTimeMBS) 534

4.67.9 setTransform(transform as CGAffineTransformMBS, time as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets a value of the transform at a time within the time range of the instruction.
Notes: transform: The transform to be applied at time.
time: A time value within the time range of the composition instruction.

Sets a fixed transform to apply from the specified time until the next time at which a transform is set. This is the same as setting a flat ramp for that time range. Before the first specified time for which a transform is set, the affine transform is held constant at the value of CGAffineTransformIdentity; after the last time for which a transform is set, the affine transform is held constant at that last value.

4.67.10 setTransformRamp(startTransform as CGAffineTransformMBS, endTransform as CGAffineTransformMBS, timeRange as CMTimeRangeMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Sets a transform ramp to apply during a given time range.
Notes: startTransform: The transform to be applied at the starting time of timeRange.
endTransform: The transform to be applied at the end time of timeRange.
timeRange: The time range over which the value of the transform is interpolated between startTransform and endTransform.

During a transform ramp, the affine transform is interpolated between the values set at the ramp's start time and end time. Before the first specified time for which a transform is set, the affine transform is held constant at the value of CGAffineTransformIdentity; after the last time for which a transform is set, the affine transform is held constant at that last value.

Available in OS X v10.7 and later.

4.67.11 videoCompositionLayerInstruction as AVMutableVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new mutable video composition layer instruction.
Notes: Returns a new mutable video composition layer instruction with no transform or opacity ramps and trackID initialized to kCMPersistentTrackID_Invalid.

4.67.12 videoCompositionLayerInstructionWithAssetTrack(track as AVAssetTrackMBS) as AVMutableVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new mutable video composition layer instruction for the given track.
Notes: track: The asset track to which to apply the instruction.

Returns a new mutable video composition layer instruction with no transform or opacity ramps and trackID initialized to the track ID of track.

4.67.13 Properties

4.67.14 trackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The trackID of the source track to which the compositor will apply the instruction.
Notes: (Read and Write computed property)

4.68 class AVMutableVideoCompositionMBS

4.68.1 class AVMutableVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVMutableVideoComposition object represents a mutable video composition.
Notes: Subclass of the AVVideoCompositionMBS class.
Blog Entries

- MBS Xojo Plugins, version 18.4pr1
- Automated video editing with Xojo
- MBS Xojo / Real Studio Plugins, version 14.5pr2

4.68.2 Methods

4.68.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a new mutable video composition.
See also:

- 4.68.4 Constructor(other as AVVideoCompositionMBS) 538

4.68.4 Constructor(other as AVVideoCompositionMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The copy constructor to initialize with a mutable copy of the given object.
See also:

- 4.68.3 Constructor 538

4.68.5 mutableVideoComposition as AVMutableVideoCompositionMBS

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Returns a new mutable video composition.

4.68.6 mutableVideoCompositionWithPropertiesOfAsset(asset as AVAssetMBS) as AVMutableVideoCompositionMBS

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Returns a new mutable video composition with the specified asset properties.
Notes: asset: An instance of AVAsset. Ensure that the duration and tracks properties of the asset are already loaded before invoking this method.
Returns a newly created and initialized instance of AVMutableVideoComposition.

The returned AVMutableVideoComposition has instructions that respect the spatial properties and time ranges of the specified asset's video tracks.

It also has the following values for its properties:

- A value for frameDuration short enough to accommodate the greatest nominalFrameRate among the asset's video tracks. If the nominalFrameRate of all of the asset's video tracks is 0, a default framerate of 30fps is used.
- If the specified asset is an instance of AVComposition, the renderSize is set to the naturalSize of the AVComposition; otherwise the renderSize will be set to a value that encompasses all of the asset's video tracks.
- A renderScale of 1.0.
- The animationTool property set to nil.

Available in OS X v10.9 and later.

4.68.7 setInstructions(items() as AVVideoCompositionInstructionMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the video composition instructions.

4.68.8 Properties

4.68.9 animationTool as AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A special video composition tool for use with Core Animation.
Notes: This attribute may be nil.

(Read and Write computed property)

4.68.10 frameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The interval for which the video composition should render composed video frames.
Notes: (Read and Write computed property)

4.68.11 renderSize as CGSizeMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The render size.
Notes: (Read and Write computed property)

4.69 class AVOutputSettingsAssistantMBS

4.69.1 class AVOutputSettingsAssistantMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The helper class to configure output settings.
Notes: A class, each instance of which specifies a set of parameters for configuring objects that use output settings dictionaries, for example AVAssetWriter & AVAssetWriterInput, so that the resulting media file conforms to some specific criteria.

Instances of AVOutputSettingsAssistant are typically created using a string constant representing a specific preset configuration, such as AVOutputSettingsPreset1280x720. Once you have an instance, its properties can be used as a guide for creating and configuring an AVAssetWriter object and one or more AVAssetWriterInput objects. If all the suggested properties are respected, the resulting media file will conform to the criteria implied by the preset. Alternatively, the properties of an instance can be used as a "base" configuration which can be customized to suit your individual needs.

The recommendations made by an instance get better as you tell it more about the format of your source data. For example, if you set the sourceVideoFormat property, the recommendation made by the videoSettings property will ensure that your video frames are not scaled up from a smaller size.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.69.2 Methods

4.69.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 or later.

4.69.4 AVOutputSettingsPreset1280x720 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the predefined presets.

4.69.5 AVOutputSettingsPreset1920x1080 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the predefined presets.

4.69.6 AVOutputSettingsPreset640x480 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the predefined presets.

4.69.7 AVOutputSettingsPreset960x540 as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the predefined presets.

4.69.8 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.

4.69.9 outputSettingsAssistantWithPreset(presetIdentifier as string) as AVOutputSettingsAssistantMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns an instance of AVOutputSettingsAssistant corresponding to the given preset.
Notes: presetIdentifier: The string identifier, for example AVOutputSettingsPreset1280x720, for the desired preset.

Returns an instance of AVOutputSettingsAssistant with properties corresponding to the given preset, or nil if there is no such preset.

The properties of the returned object can be used as a guide for creating and configuring an AVAssetWriter object and one or more AVAssetWriterInput objects. If all the suggested properties are respected in creating the AVAssetWriter, the resulting media file will conform to the criteria implied by the preset.

4.69.10 Properties

4.69.11 audioSettings as Dictionary

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A dictionary of key/value pairs, as specified in AVAudioSettings.h, to be used when e.g. creating an instance of AVAssetWriterInput.
Notes: The value of this property may change as a result of setting a new value for the sourceAudioFormat property.
(Read only property)

4.69.12 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.69.13 outputFileType as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A UTI indicating the type of file to be written, to be used when e.g. creating an instance of AVAssetWriter.
Notes: Use UTTypeCopyPreferredTagWithClass / kUTTagClassFilenameExtension to get a suitable file extension for a given file type.
(Read only property)

4.69.14 videoSettings as Dictionary

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A dictionary of key/value pairs, as specified in AVVideoSettings.h, to be used when e.g. creating an instance of AVAssetWriterInput.
Notes: The value of this property may change as a result of setting a new value for the sourceVideoFormat property.
(Read only property)

4.70 class AVPixelAspectRatioMBS

4.70.1 class AVPixelAspectRatioMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The class for pixel aspect ratio.

4.70.2 Properties

4.70.3 horizontalSpacing as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The horizontal spacing of the pixelAspectRatio property.
Notes: (Read and Write property)

4.70.4 verticalSpacing as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The vertical spacing of the pixelAspectRatio property.
Notes: (Read and Write property)

4.71 class AVPlayerItemAccessLogEventMBS

4.71.1 class AVPlayerItemAccessLogEventMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVPlayerItemAccessLogEvent object represents a single item in an AVPlayerItem object's access log.
Notes: An AVPlayerItemAccessLog object provides named properties for accessing the data fields of each log event. None of the properties of this class are observable using key-value observing.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.71.2 Methods

4.71.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.71.4 Properties

4.71.5 downloadOverdue as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The total number of times the download of the segments took too long. (read-only)
Notes: The value of the property is negative if unknown.
This property corresponds to "c-overdue".
This property is not compatible with key-value observing.
Available in OS X v10.9 and later.
(Read only property)

4.71.6 durationWatched as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The accumulated duration of the media played, in seconds. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.7 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.71.8 indicatedBitrate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The throughput required to play the stream, as advertised by the server, in bits per second. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.9 mediaRequestsWWAN as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Number of network read requests over WWAN. (read-only)
Notes: The value of the property is negative if unknown.
Corresponds to "sc-wwan-count".
This property is not compatible with key-value observing.
Available in OS X v10.9 and later.
(Read only property)

4.71.10 numberOfBytesTransferred as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The accumulated number of bytes transferred by the item. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.11 numberOfDroppedVideoFrames as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The total number of dropped video frames (read-only)
Notes: The value of this property is negative if unknown.

(Read only property)

4.71.12 numberOfMediaRequests as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A count of media read requests from the server to this client. (read-only)
Notes: For HTTP live streaming, this property contains the count of media requests downloaded from the server. For progressive-style HTTP media downloads, it contains a count of HTTP GET (byte-range) requests for the resource.
The value of this property is negative if unknown.
Available in OS X v10.9 and later.#
(Read only property)

4.71.13 numberOfSegmentsDownloaded as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A count of the media segments downloaded from the server to this client. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.14 numberOfServerAddressChanges as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A count of changes to the server address over the last uninterrupted period of playback. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.15 numberOfStalls as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The total number of playback stalls encountered. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.16 observedBitrate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The empirical throughput across all media downloaded, in bits per second. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.17 observedBitrateStandardDeviation as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Standard deviation of observed segment download bit rates. (read-only)
Notes: The value of the property is negative if unknown.
Available in OS X v10.9 and later.
(Read only property)

4.71.18 observedMaxBitrate as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Maximum observed segment download bit rate. (read-only)
Notes: The value of the property is negative if unknown.
Available in OS X v10.9 and later.#
(Read only property)

4.71.19 observedMinBitrate as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Minimum observed segment download bit rate. (read-only)
Notes: The value of the property is negative if unknown.
Available in OS X v10.9 and later.
(Read only property)

4.71.20 playbackSessionID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A GUID that identifies the playback session. (read-only)
Notes: This value is used in HTTP requests.
The value of this property is nil if unknown.

(Read only property)

4.71.21 playbackStartDate as date

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time at which playback began for this event. (read-only)
Notes: The value of this property is nil if unknown.
(Read only property)

4.71.22 playbackStartDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The date and time at which playback began for this event. (read-only)
Notes: The value of this property is nil if unknown.
(Read only property)

4.71.23 playbackStartOffset as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An offset into the playlist where the last uninterrupted period of playback began, in seconds (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.24 playbackType as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The playback type: live, VOD, or from a file. (read-only)
Notes: If nil is returned the playback type is unknown.
Available in OS X v10.9 and later.
(Read only property)

4.71.25 segmentsDownloadedDuration as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The accumulated duration of the media downloaded, in seconds. (read-only)
Notes: The value of this property is negative if unknown.
(Read only property)

4.71.26 serverAddress as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The IP address of the server that was the source of the last delivered media segment. (read-only)
Notes: The value of this property is nil if unknown.
(Read only property)

4.71.27 startupTime as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The accumulated duration, in seconds, until player item is ready to play. (read-only)
Notes: The value of the property is negative if unknown.
Available in OS X v10.9 and later.
(Read only property)

4.71.28 switchBitrate as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Bandwidth that caused a switch (up or down). (read-only)
Notes: The value of the property is negative if unknown.
Available in OS X v10.9 and later.
(Read only property)

4.71.29 transferDuration as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The accumulated duration, in seconds, of active network transfer of bytes. (read-only)
Notes: The value of the property is negative if unknown.
Corresponds to ”c-transfer-duration”.
This property is not compatible with key-value observing.
Available in OS X v10.9 and later.
(Read only property)

4.71.30 URI as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URI of the playback item (read-only)
Notes: The value of this property may be nil if the URI is unknown.
(Read only property)

4.72 class AVPlayerItemAccessLogMBS

4.72.1 class AVPlayerItemAccessLogMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVPlayerItemAccessLog object to retrieve the access log associated with an AVPlayerItem object.
Notes: An AVPlayerItemAccessLog object accumulates key metrics about network playback and presents them as a collection of AVPlayerItemAccessLogEvent instances. Each event instance collates the data that relates to each uninterrupted period of playback.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.72.2 Methods

4.72.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.72.4 events as AVPlayerItemAccessLogEventMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A chronologically ordered array of AVPlayerItemAccessLogEvent objects. (read-only)
Notes: The array contains AVPlayerItemAccessLogEvent objects that represent the chronological sequence of events contained in the access log.

This property is not observable using key-value observing.

4.72.5 extendedLogData as MemoryBlock

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a serialized representation of the access log in the Extended Log File Format.
Notes: A serialized representation of the access log in the Extended Log File Format.

This method converts the web server access log into a textual format that conforms to the W3C Extended Log File Format for web server log files. For more information, see http://www.w3.org/pub/WWW/TR/WD-logfile.html.

4.72.6 extendedLogDataStringEncoding as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the string ecoding of the extended log data.
Notes: The string encoding of the data returned by extendedLogData.

4.72.7 Properties

4.72.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.73 class AVPlayerItemErrorLogEventMBS

4.73.1 class AVPlayerItemErrorLogEventMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVPlayerItemErrorLogEvent object represents a single item in an AVPlayerItem object's error log.
Notes: An AVPlayerItemErrorLogEvent object provides named properties for accessing the data fields of each log event. None of the properties of this class are observable using key-value observing.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.73.2 Methods

4.73.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.73.4 date as date

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date and time when the error occurred. (read-only)
Notes: The property corresponds to "date".
The value of this property may be nil if the date is unknown.

4.73.5 dateTime as dateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The date and time when the error occurred. (read-only)
Notes: The property corresponds to "date".
The value of this property may be nil if the date is unknown.

4.73.6 errorComment as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A description of the error encountered (read-only)
Notes: The value of this property may be nil if further information is not available.

4.73.7 errorDomain as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The domain of the error. (read-only)

4.73.8 errorStatusCode as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A unique error code identifier. (read-only)

4.73.9 playbackSessionID as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A GUID that identifies the playback session. (read-only)
Notes: The value of this property is used in HTTP requests, and may be nil if the GUID is unknown.

4.73.10 serverAddress as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The IP address of the server that was the source of the error. (read-only)
Notes: The value of this property can be either an IPv4 or IPv6 address, and may be nil if the address is unknown.

4.73.11 URI as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URI of the playback item (read-only)
Notes: The value of this property may be nil if the URI is unknown.

4.73.12 Properties

4.73.13 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.74 class AVPlayerItemErrorLogMBS

4.74.1 class AVPlayerItemErrorLogMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVPlayerItemErrorLog object to retrieve the error log associated with an AVPlayerItem object.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.74.2 Methods

4.74.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.74.4 events as AVPlayerItemErrorLogEventMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A chronologically ordered array of AVPlayerItemErrorLogEvent objects. (read-only)
Notes: The array contains AVPlayerItemErrorLogEvent objects that represent the chronological sequence of events contained in the error log.

This property is not observable using key-value observing.

4.74.5 extendedLogData as MemoryBlock

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a serialized representation of the error log in the Extended Log File Format.
Notes: A serialized representation of the error log in the Extended Log File Format.

Discussion
This method converts the web server error log into a textual format that conforms to the W3C Extended Log File Format for web server log files. For more information, see http://www.w3.org/pub/WWW/TR/WD-logfile.html.

4.74.6 extendedLogDataStringEncoding as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the string ecoding of the extended log data.
Notes: The string encoding of the data returned by extendedLogData.

4.74.7 Properties

4.74.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.75 class AVPlayerItemLegibleOutputMBS

4.75.1 class AVPlayerItemLegibleOutputMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A subclass of AVPlayerItemOutput that can vend media with a legible characteristic as NSAttributedStrings.
Notes: Subclass of the AVPlayerItemOutputMBS class.

4.75.2 Methods

4.75.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 or later.

4.75.4 AVPlayerItemLegibleOutputTextStylingResolutionDefault as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the text styling mode.
Notes: Specify this level of text styling resolution to receive attributed strings from an AVPlayerItemLegibleOutput that include the same level of styling information that AVFoundation would use itself to render text within an AVPlayerLayer. The text styling will accommodate user-level Media Accessibility settings.

4.75.5 AVPlayerItemLegibleOutputTextStylingResolutionSourceAndRulesOnly as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: One of the text styling mode.
Notes: Specify this level of text styling resolution to receive only the styling present in the source media and the styling provided via AVPlayerItem.textStyleRules.
This level of resolution excludes styling provided by the user-level Media Accessibility settings. You would typically use it if you wish to override the styling specified in source media. If you do this, you are strongly encouraged to allow your custom styling in turn to be overriden by user preferences for text styling that are available as Media Accessibility settings.

4.75.6 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The standard constructor.
See also:

- 4.75.7 Constructor(subTypes() as string) 560

4.75.7 Constructor(subTypes() as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The constructor.
Notes: Returns an instance of AVPlayerItemLegibleOutput with filtering enabled for AVPlayerItemLegibleOutputPushDelegate's legibleOutput:didOutputAttributedStrings:nativeSampleBuffers:forItemTime:.

subtypes: Array of strings with 4 letter codes.

Add media subtype FourCC number objects to the subtypes array to elect to receive that type as a CMSampleBuffer instead of an NSAttributedString. Initializing an AVPlayerItemLegibleOutput using the Constructor method is equivalent to calling Constructor with an empty array, which means that all legible data, regardless of media subtype, will be delivered using NSAttributedString in a common format.

If a media subtype for which there is no legible data in the current player item is included in the media subtypes array, no error will occur. AVPlayerItemLegibleOutput will not vend closed caption data as CMSampleBuffers, so it is an error to include 'c608' in the media subtypes array.
See also:

- 4.75.6 Constructor 560

4.75.8 Properties

4.75.9 advanceIntervalForDelegateInvocation as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Permits advance invocation of the associated delegate, if any.
Notes: If it is possible, an AVPlayerItemLegibleOutput will message its delegate advanceIntervalForDelegateInvocation seconds earlier than otherwise. If the value you provide is large, effectively requesting provision of samples earlier than the AVPlayerItemLegibleOutput is prepared to act on them, the delegate will be invoked as soon as possible.
(Read and Write property)

4.75.10 textStylingResolution as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A string identifier indicating the degree of text styling to be applied to attributed strings vended by the receiver.
Notes: Valid values are AVPlayerItemLegibleOutputTextStylingResolutionDefault and AVPlayerItemLegibleOutputTextStylingResolutionSourceAndRulesOnly. An NSInvalidArgumentException is raised if this property is set to any other value. The default value is AVPlayerItemLegibleOutputTextStylingResolutionDefault, which indicates that attributed strings vended by the receiver will include the same level of styling information that would be used if AVFoundation were rendering the text via AVPlayerLayer.
(Read and Write property)

4.76 class AVPlayerItemMBS

4.76.1 class AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVPlayerItem represents the presentation state of an asset that's played by an AVPlayer object, and lets you observe that state.
Notes: A object carries a reference to an AVAsset object and presentation settings for that asset, including track enabled state. If you need to inspect the media assets themselves, you should message the AVAsset object itself.

You can initialize a player item using an URL (playerItemWithURL and Constructor); the resource types referenced by the URL may include, but aren't necessarily limited to, those with the following corresponding UTIs:

kUTTypeQuickTimeMovie, (.mov, .qt)
kUTTypeMPEG4 (.mp4)
"public.3gpp" (.3gp, .3gpp)
kUTTypeMPEG4Audio (.m4a)
"com.apple.coreaudio-format" (.caf)
"com.microsoft.waveform-audio" (.wav)
"public.aiff-audio" (.aif)
"public.aifc-audio" (also .aif)
"org.3gpp.adaptive-multi-rate-audio" (.amr)

If you want to play an asset more than once within a sequence of items, you must create independent instances of AVPlayerItem for each placement in the player's queue.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3

4.76.2 Methods

4.76.3 accessLog as AVPlayerItemAccessLogMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an object that represents a snapshot of the network access log.
Notes: An object that represents a snapshot of the network access log. The returned value can be nil.
If the method returns nil, there is no logging information currently available for the player item.

4.76.4 addOutput(output as AVPlayerItemOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Adds the specified player item output object to the receiver.
Notes: output: The player item output object to associate with the item.

When you add an AVPlayerItemOutput object to an item, the samples associated with that output object are processed according to the rules for mixing, composing, or excluding content that the AVPlayer object honors for the specific media type. For example, video media is composed according to the instructions provided by the player item's video composition object and audio media is mixed according to the parameters of its audio mix object.
Available in OS X v10.8 and later.

4.76.5 automaticallyLoadedAssetKeys as String()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array of property keys defined by the asset property. (read-only)
Notes: The value of each key in automaticallyLoadedAssetKeys will be automatically be loaded by the asset before the receiver achieves the status AVPlayerItemStatusReadyToPlay; i.e. when the item is ready to play, the value of invoking statusOfValueForKey on the asset property value will be AVKeyValueStatusLoaded. If loading of any of the values fails, the status property of the receiver will change to AVPlayerItemStatus-Failed.
Available in OS X v10.9 and later.

4.76.6 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.76.7 cancelContentAuthorizationRequest

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Causes the currently outstanding content authorization request to be cancelled.
Notes: Calling this method while a content authorization request is pending will cause that request to be cancelled and its completion handler to be invoked with a status of AVContentAuthorizationCancelled.

This method does not block.

4.76.8 cancelPendingSeeks

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancel any pending seek requests and invoke the corresponding completion handlers if present.
Notes: Use this method to cancel and release the completion handlers of pending seeks.
The finished parameter of the completion handlers will be set to false.

4.76.9 Constructor(asset as AVAssetMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a new player item for a given asset.
See also:

- 4.76.10 Constructor(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string) 564
- 4.76.11 Constructor(file as folderitem) 565
- 4.76.12 Constructor(URL as string) 565

4.76.10 Constructor(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Initializes a player item with an array of AVAsset keys.
Notes: asset: An instance of AVAsset.
automaticallyLoadedAssetKeys: An array of strings, each representing a property defined by AVAsset.

Returns an initialized instance of AVPlayerItem.

The value of each key in automaticallyLoadedAssetKeys will be automatically be loaded by the underlying AVAsset before the receiver achieves the status AVPlayerItemStatusReadyToPlay; i.e. when the item is ready to play, the value returned by invoking the asset property's statusOfValueForKey method will be one of the terminal status values, either AVKeyValueStatusLoaded, AVKeyValueStatusFailed, or AVKeyValueStatusCancelled.

Important: The asset property keys "playable" and "compatibleWithSavedPhotosAlbum" are not eligible for automatic loading by AVPlayerItem. You must use the AVAsset method loadValuesAsynchronously-

ForKeys:completionHandler: to load the values of those properties asynchronously.

Available in OS X v10.9 and later.
See also:

- 4.76.9 Constructor(asset as AVAssetMBS) 564
- 4.76.11 Constructor(file as folderitem) 565
- 4.76.12 Constructor(URL as string) 565

4.76.11 Constructor(file as folderitem)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prepares a player item with a given file.
Notes: file: A folderitem.

This method immediately returns the item, but with the status AVPlayerItemStatusUnknown.
If the URL contains valid data that can be used by the player item, the status later changes to AVPlayerItemStatusReadyToPlay.
If the URL contains no valid data or otherwise can't be used by the player item, the status later changes to AVPlayerItemStatusFailed.
See also:

- 4.76.9 Constructor(asset as AVAssetMBS) 564
- 4.76.10 Constructor(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string) 564
- 4.76.12 Constructor(URL as string) 565

4.76.12 Constructor(URL as string)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prepares a player item with a given URL.
Notes: URL: An URL.

This method immediately returns the item, but with the status AVPlayerItemStatusUnknown.
If the URL contains valid data that can be used by the player item, the status later changes to AVPlayerItemStatusReadyToPlay.
If the URL contains no valid data or otherwise can't be used by the player item, the status later changes to AVPlayerItemStatusFailed.
See also:

- 4.76.9 Constructor(asset as AVAssetMBS) 564

- 4.76.10 Constructor(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string) 564
- 4.76.11 Constructor(file as folderitem) 565

4.76.13 copy as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.76.14 errorLog as AVPlayerItemErrorLogMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an object that represents a snapshot of the error log.
Notes: An object that represents a snapshot of the error log. The returned value can be nil.
If the method returns nil, there is no logging information currently available for the player item.

4.76.15 loadedTimeRanges as CMTimeRangeMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time ranges of the item that have been loaded. (read-only)

4.76.16 outputs as AVPlayerItemOutputMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The outputs associated with the item. (read-only)
Notes: This property contains the collection of AVPlayerItemOutput objects used to transfer media data to the player object.
Available in OS X v10.8 and later.

4.76.17 playerItemWithAsset(asset as AVAssetMBS) as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player item for a given asset.
See also:

- 4.76.18 playerItemWithAsset(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string) as AVPlayerItemMBS 567

4.76.18 playerItemWithAsset(asset as AVAssetMBS, automaticallyLoadedAssetKeys() as string) as AVPlayerItemMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates and initializes a player item with an array of AVAsset keys.
Notes: asset: An instance of AVAsset.
automaticallyLoadedAssetKeys: An array of strings, each representing a property key defined by AVAsset.
Returns an initialized instance of AVPlayerItem.

The value of each key in automaticallyLoadedAssetKeys will be automatically be loaded by the underlying AVAsset before the receiver achieves the status AVPlayerItemStatusReadyToPlay; i.e. when the item is ready to play, the value returned by invoking the asset property's statusOfValueForKey:error: method will be one of the terminal status values, either AVKeyValueStatusLoaded, AVKeyValueStatusFailed, or AVKeyValueStatusCancelled.

Important: The asset property keys "playable" and "compatibleWithSavedPhotosAlbum" are not eligible for automatic loading by AVPlayerItem. You must use the AVAsset method loadValuesAsynchronouslyForKeys:completionHandler: to load the values of those properties asynchronously.

Available in OS X v10.9 and later.
See also:

- 4.76.17 playerItemWithAsset(asset as AVAssetMBS) as AVPlayerItemMBS 566

4.76.19 playerItemWithFile(file as folderitem) as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player item, prepared to use a given folderitem.
Notes: File: A folderitem.
Returns a new player item, prepared to use folderitem.

This method immediately returns the item, but with the status AVPlayerItemStatusUnknown.
If the URL contains valid data that can be used by the player item, the status later changes to AVPlayerItemStatusReadyToPlay.
If the URL contains no valid data or otherwise can't be used by the player item, the status later changes to AVPlayerItemStatusFailed.

4.76.20 playerItemWithURL(URL as string) as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player item, prepared to use a given URL.
Notes: URL: An URL.
Returns a new player item, prepared to use URL.

This method immediately returns the item, but with the status AVPlayerItemStatusUnknown.
If the URL contains valid data that can be used by the player item, the status later changes to AVPlayerItemStatusReadyToPlay.
If the URL contains no valid data or otherwise can't be used by the player item, the status later changes to AVPlayerItemStatusFailed.

4.76.21 removeOutput(output as AVPlayerItemOutputMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes the specified player item output object from the receiver.
Notes: output: The player item output object to remove.
Available in OS X v10.8 and later.

4.76.22 requestContentAuthorizationAsynchronously(timeoutInterval as Double, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Causes appropriate action to be taken to allow the user to authorize the content for playback.
Notes: timeoutInterval: The maximum amount of time in seconds to wait for the user to authorize the content before calling the handler block with a timeout result.

Calling this method will present the user with the opportunity to authorize the content (for example, by launching iTunes and prompting the user to enter their Apple ID and password).

When the user has taken action (or the timeout has elapsed), the AVFoundationMBS.requestContentAuthorizationCompleted event is invoked. You determine the status of the authorization attempt by checking the value of the contentAuthorizationRequestStatus property.

Even if the status indicates a completed authorization, the content may still not be authorized (for example, if the user authorizes an Apple ID other than that associated with the content). You should re-check the value of contentAuthorizationRequestStatus to verify whether the content has actually been authorized before continuing. It is not necessary to call this method if the value of contentAuthorizationRequestStatus is already true.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.76.23 seekableTimeRanges as CMTimeRangeMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of time ranges within which it is possible to seek. (read-only)

4.76.24 seekToDate(date as date, fireEvent as boolean = false, tag as Variant = nil) as boolean

Plugin Version: 14.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Moves the playback cursor to the time given by the specified date object.
Notes: date: The time to which to seek.

If fireEvent is true, the AVFoundationMBS.playerItemSeekToDateFinished event is called.

Use this method to seek to a specified time in the item item and be notified when the operation completes. If the seek request completes without being interrupted (either by another seek request or by any other operation), the completion handler you provide is executed with the finished parameter set to true.

If another seek request is already in progress when you call this method, the completion handler for the in-progress seek request is executed immediately with the finished parameter set to NO.
Available in OS X v10.9 and later.

4.76.25 seekToDateTime(date as dateTime, fireEvent as boolean = false, tag as variant = nil) as boolean

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Moves the playback cursor to the time given by the specified date object.
Notes: date: The time to which to seek.

If fireEvent is true, the AVFoundationMBS.playerItemSeekToDateFinished event is called.

Use this method to seek to a specified time in the item item and be notified when the operation completes. If the seek request completes without being interrupted (either by another seek request or by any other operation), the completion handler you provide is executed with the finished parameter set to true.

If another seek request is already in progress when you call this method, the completion handler for the in-progress seek request is executed immediately with the finished parameter set to NO.
Available in OS X v10.9 and later.

4.76.26 seekToTime(time as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Moves the playback cursor to a given time.
Notes: time: The time to which to move the playback cursor.
If fireEvent is true, the AVFoundationMBS.playerItemSeekToTimeFinished event will be called later.

Use this method to seek to a specified time in the item item and be notified when the operation completes. If the seek request completes without being interrupted (either by another seek request or by any other operation), the event is executed with the finished parameter set to true.

If another seek request is already in progress when you call this method, the completion handler for the in-progress seek request is executed immediately with the finished parameter set to false.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.76.27 seekToTime(time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil) 570

4.76.27 seekToTime(time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Moves the playback cursor within a specified time bound.
Notes: time: The time to which you would like to move the playback cursor.
toleranceBefore: The tolerance allowed before time.
toleranceAfter: The tolerance allowed after time.

The time seeked to will be within the range [time-beforeTolerance, time+afterTolerance] , and may differ from the specified time for efficiency. If you pass kCMTimeZero for both toleranceBefore and toleranceAfter (to request sample accurate seeking), you may incur additional decoding delay.

Passing kCMTimePositiveInfinity for both toleranceBefore and toleranceAfter is the same as messaging seekToTime directly.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.76.26 seekToTime(time as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil) 570

4.76.28 selectedMediaOptionInMediaSelectionGroup(mediaSelectionGroup as AVMediaSelectionGroupMBS) as AVMediaSelectionOptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the media selection option that's currently selected from the specified group.
Notes: mediaSelectionGroup: A media selection group obtained from the player item's asset.

Returns an instance of AVMediaSelectionOption that describes the currently selection option in the group.

If the value of the allowsEmptySelection property of mediaSelectionGroup is true, the currently selected option in the group may be nil.
Available in OS X v10.8 and later.

4.76.29 selectMediaOption(mediaSelectionOption as AVMediaSelectionOptionMBS, mediaSelectionGroup as AVMediaSelectionGroupMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Selects the media option described by a specified instance of AVMediaSelectionOption in a given media selection group and deselects all other options in that group.
Notes: mediaSelectionOption: The option to select.
If the value of the allowsEmptySelection property of mediaSelectionGroup is true, you can pass nil to deselect all media selection options in the group.
mediaSelectionGroup: The media selection group, obtained from the receiver's asset, that contains mediaSelectionOption.

If mediaSelectionOption isn't a member of the mediaSelectionGroup, no change in presentation state will result.

If multiple options within a group meet your criteria for selection according to locale or other considerations, and if these options are otherwise indistinguishable to you according to media characteristics that are meaningful for your application, content is typically authored so that the first available option that meets your criteria is appropriate for selection.
Available in OS X v10.8 and later.

4.76.30 selectMediaOptionAutomaticallyInMediaSelectionGroup(mediaSelectionGroup as AVMediaSelectionGroupMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Selects the media option in the specified media selection group that best matches the receiver's automatic selection criteria.
Notes: mediaSelectionGroup: The media selection group, obtained from the receiver's asset, that contains the specified option.

This method has no effect unless the appliesMediaSelectionCriteriaAutomatically property of the associated AVPlayer is true and unless automatic media selection has previously been overridden by invoking sselectMediaOption:inMediaSelectionGroup.
Available in OS X v10.9 and later.

4.76.31 setTextStyleRules(rules() as AVTextStyleRuleMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Sets the textStyleRules array.

4.76.32 stepByCount(stepCount as Integer)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Moves the player's current item's current time forward or backward by a specified number of steps.
Notes: stepCount: The number of steps by which to move.
A positive number steps forward, a negative number steps backward.

The size of each step depends on the receiver's enabled AVPlayerItemTrack objects (see tracks).

4.76.33 textStyleRules as AVTextStyleRuleMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array of text style rules to apply to subtitles and other legible text.
Notes: You can use this property to assign an array of AVTextStyleRule objects to the item. Each rule specifies both the style information and the range of text to which that styling should apply.
Available in OS X v10.9 and later.

4.76.34 timedMetadata as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The timed metadata played most recently by the media stream. (read-only)
Notes: The array contains instances of AVMetadataItem.

4.76.35 tracks as AVPlayerItemTrackMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An array of AVPlayerItemTrack objects. (read-only)
Notes: This property can change dynamically during playback.
You can observe this property using key-value observing.

4.76.36 Properties

4.76.37 asset as AVAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The underlying asset provided during initialization. (read-only)
Notes: (Read only property)

4.76.38 audioMix as AVAudioMixMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The audio mix parameters to be applied during playback.
Notes: (Read and Write property)

4.76.39 audioTimePitchAlgorithm as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The processing algorithm used to manage audio pitch for scaled audio edits.
Notes: The supported constants are defined in Time Pitch Algorithm Settings.
An NSInvalidArgumentException will be raised if this property is set to a value other than the defined constants.
Available in OS X v10.9 and later.
(Read and Write property)

4.76.40 canPlayFastForward as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item can be played at rates greater than 1.0. (read-only)
Notes: Available in OS X v10.8 and later.
(Read only property)

4.76.41 canPlayFastReverse as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item can be played at rates less than –1.0. (read-only)
Notes: Available in OS X v10.8 and later.
(Read only property)

4.76.42 canPlayReverse as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item can be played with a rate of -1.0. (read-only)
Notes: Available in OS X v10.8 and later.
(Read only property)

4.76.43 canPlaySlowForward as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item can be played at a rate between 0.0 and 1.0. (read-only)
Notes: Available in OS X v10.8 and later.

(Read only property)

4.76.44 canPlaySlowReverse as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item can be played at a rate between -1.0 and 0.0. (read-only)
Notes: Available in OS X v10.8 and later.
(Read only property)

4.76.45 canStepBackward as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item supports stepping backward. (read-only)
Notes: Once the item becomes ready to play, the value of this property does not change. This behavior applies even when boundary conditions, such as when the item's current time is kCMTimeZero, have been reached.
Available in OS X v10.8 and later.
(Read only property)

4.76.46 canStepForward as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item supports stepping forward. (read-only)
Notes: Once the item becomes ready to play, the value of this property does not change. This behavior applies even when boundary conditions, such as when the item's current time is equal to its end time, have been reached.
Available in OS X v10.8 and later.
(Read only property)

4.76.47 contentAuthorizationRequestStatus as Integer

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates the status of the most recent call to requestContentAuthorizationAsynchronouslyWithTimeoutInterval. (read-only)
Notes: This property reports the authorization status as determined by the most recent call to requestContentAuthorizationAsynchronouslyWithTimeoutInterval.

The value will be AVContentAuthorizationUnknown before the first call and between the time a request call is made and just prior to the completion handler being executed (thus it is safe to query this property from the event).

This value is not key-value observable.
(Read only property)

4.76.48 currentDate as date

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the current time of the item as a date object.
Notes: The current time of the item as a date object, or nil if playback is not mapped to any date.
(Read only property)

4.76.49 currentDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Returns the current time of the item as a date object.
Notes: The current time of the item as a date object, or nil if playback is not mapped to any date.
(Read only property)

4.76.50 currentTime as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the current time of the item.
Notes: (Read only property)

4.76.51 customVideoCompositor as AVVideoCompositingMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The custom video compositor, if any. (read-only)
Notes: The custom video compositor instance that is used during image generation is accessible via this property after the value of AVPlayerItemStatusFailed is set to an AVVideoComposition instance that specifies a custom video compositor class. Any additional communication between the application and that instance of the custom video compositor, if any is required for configuration or other purposes, can only

occur once that has happened.

If the value of AVPlayerItemStatusFailed is changed from an AVVideoComposition that specifies a custom video compositor class to another instance of AVVideoComposition that specifies the same custom video compositor class, the instance of the custom video compositor that was previously created will receive the renderContextChanged: message and remain in use for subsequent image generation.

This property is nil if there is no video compositor, or if the internal video compositor is in use.
Available in OS X v10.9 and later.
(Read only property)

4.76.52 duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the duration of the item. (read-only)
Notes: Indicates the duration of the item, not considering either its forwardPlaybackEndTime or reversePlaybackEndTime.
(Read only property)

4.76.53 error as NSErrorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: If the receiver's status is AVPlayerItemStatusFailed, this describes the error that caused the failure. (read-only)
Notes: The value of this property is an error that describes what caused the receiver to no longer be able to be played.
If the receiver's status is not AVPlayerItemStatusFailed, the value of this property is nil.
(Read only property)

4.76.54 forwardPlaybackEndTime as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time at which forward playback ends.
Notes: The value indicated the time at which playback should end when the playback rate is positive (see AVPlayer's rate property).

The default value is kCMTimeInvalid, which indicates that no end time for forward playback is specified. In this case, the effective end time for forward playback is the item's duration.

The value of this property has no effect on playback when the rate is negative.
(Read and Write property)

4.76.55 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.76.56 isApplicationAuthorizedForPlayback as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the application can be used to play the content. (read-only)
Notes: This property reports whether or not the calling application is authorized to play the content associated with the item.

Application authorization is independent of content authorization (see contentAuthorizedForPlayback) and that both must be granted in order for an application to be allowed to play protected content. Also, unlike content authorization, application authorization is not dependent on user credentials (that is, if applicationAuthorizedForPlayback is false, there are no means to obtain authorization).

This property is not key-value observable.
(Read only property)

4.76.57 isAuthorizationRequiredForPlayback as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether or not authorization is required to play the content. (read-only)
Notes: This property reports whether authorization is required for the item's content to be played. If it does not require authorization, then none of the other authorization-related methods or properties apply (though they will return sensible values where possible).

This property is not key-value observable.
(Read only property)

4.76.58 isContentAuthorizedForPlayback as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the content has been authorized by the user (for example, by authorizing the content's associated account in iTunes). (read-only)
Notes: This property reports whether the user has provided the necessary credentials to the system in order for the content to be decrypted for playback.

Content authorization is independent of application authorization (see applicationAuthorizedForPlayback) and that both must be granted in order for an application to be allowed to play protected content.

This property is not key-value observable.
(Read only property)

4.76.59 isPlaybackBufferEmpty as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether playback has consumed all buffered media and that playback will stall or end. (read-only)
Notes: (Read only property)

4.76.60 isPlaybackBufferFull as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the internal media buffer is full and that further I/O is suspended. (read-only)
Notes: Despite the playback buffer reaching capacity there might not exist sufficient statistical data to support a playbackLikelyToKeepUp prediction of true.
(Read only property)

4.76.61 isPlaybackLikelyToKeepUp as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the item will likely play through without stalling (read-only)
Notes: This property communicates a prediction of playability. Factors considered in this prediction include I/O throughput and media decode performance. It is possible for playbackLikelyToKeepUp to indicate false while the property playbackBufferFull indicates true. In this event the playback buffer has reached capacity but there isn't the statistical data to support a prediction that playback is likely to keep up in the future. It is up to you to decide whether to continue media playback.

(Read only property)

4.76.62 presentationSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The size at which the visual portion of the item is presented by the player. (read-only)
Notes: You can scale the presentation size to fit within the bounds of a player layer using its videoGravity property. You can also scale the presentation size arbitrarily using the frame property of an AVPlayerLayer object.
(Read only property)

4.76.63 reversePlaybackEndTime as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time at which reverse playback ends.
Notes: The value indicated the time at which playback should end when the playback rate is negative (see AVPlayer's rate property).

The default value is kCMTimeInvalid, which indicates that no end time for reverse playback is specified. In this case, the effective end time for reverse playback is kCMTimeZero.

The value of this property has no effect on playback when the rate is positive.
(Read and Write property)

4.76.64 seekingWaitsForVideoCompositionRendering as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the item's timing follows the displayed video frame when seeking with a video composition.
Notes: By default, item timing is updated as quickly as possible during seeking. Specifically, the item does not wait for new frames to be rendered when seeking during normal playback. In most situations, the latency between the completion of a seek operation and the display of a video frame at the new time is negligible. However, when video compositions are in use, the processing of video may introduce noticeable latency. Setting the value of this property to true causes the item's timing to be updated only after the corresponding video frame has been displayed. For example, this allows an AVSynchronizedLayer object associated with the item to remain in sync with the displayed video.

This property has no effect on items whose videoComposition property is nil.

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

4.76.65 status as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The status of the player item. (read-only)
Notes: For example, whether the item is playable. For values: AVPlayerItemStatusUnknown, AVPlayerItemStatusReadyToPlay and AVPlayerItemStatusFailed.
(Read only property)

4.76.66 videoComposition as AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video composition settings to be applied during playback.
Notes: (Read and Write property)

4.76.67 Constants

Authorization Status

Constant	Value	Description
AVContentAuthorizationBusy	4	The last call to request content authorization could not be completed be another asset is currently attempting authorization.
AVContentAuthorizationCancelled	2	The last call to request content authorization was cancelled by the user.
AVContentAuthorizationCompleted	1	The last completed call to request content authorization completed.
AVContentAuthorizationNotAvailable	5	The last call to request content authorization could not be completed be there was no known mechanism by which to attempt authorization.
AVContentAuthorizationNotPossible	6	The last call to request content authorization could not be completed non-recoverable way (for example, a newer version of iTunes is required).
AVContentAuthorizationTimedOut	3	The last call to request content authorization was cancelled because the tim interval was reached.
AVContentAuthorizationUnknown	0	No call to request content authorization has completed yet.

Status Constants

Constant	Value	Description
AVPlayerItemStatusFailed	2	The item cannot be played.
AVPlayerItemStatusReadyToPlay	1	The item is ready to play.
AVPlayerItemStatusUnknown	0	The item's status is unknown.

4.77 class AVPlayerItemOutputMBS

4.77.1 class AVPlayerItemOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The AVPlayerItemOutput class is an abstract class that defines the common interface for moving samples from an asset to an AVPlayer object.
Notes: You do not create instances of this class directly but instead use one of the concrete subclasses that manage specific types of assets.

This class provides basic methods for converting time values to the timebase of the item. It also provides an option to suppress rendering of the output associated with the specific instance of this class.

Available in OS X v10.8 and later.

4.77.2 Methods

4.77.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.77.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.

4.77.5 itemTimeForHostTime(hostTimeInSeconds as Double) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Converts a host time (specified in seconds) to the item's timebase.
Notes: hostTimeInSeconds: A host time value, specified in seconds. For example, you might specify the time value returned by the CACurrentMediaTime function or the timestamp from a CADisplayLink object for this parameter.

Returns the equivalent time in the item's timebase.

The timestamp associated with a CADisplayLink object represents the time of the most recent screen refresh, which is usually a time in the past. If you want to find the time associated with the next screen refresh, you need to increment the timestamp by the value in the display link's duration property.

4.77.6 itemTimeForMachAbsoluteTime(machAbsoluteTime as Int64) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Converts a Mach host time to the item's timebase.
Notes: machAbsoluteTime: The Mach host time to convert. You typically retrieve this value using the mach_absolute_time function.

Returns the equivalent time in the item's timebase.
Available in OS X v10.8 and later.

4.77.7 Properties

4.77.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.77.9 suppressesPlayerRendering as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Boolean indicating whether the player object renders the receiver's output.
Notes: When the value of this property is false (the default), the player object handles the rendering of the receiver's associated output. You can change the value of this property to true if you want to suppress the rendering of the media data associated with this object.
Available in OS X v10.8 and later.
(Read and Write property)

4.78 class AVPlayerItemTrackMBS

4.78.1 class AVPlayerItemTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The track of a player item.
Notes: You use an AVPlayerItemTrack object to modify the presentation state of an asset track (AVAsset-Track) being presented by an AVPlayer object.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.78.2 Methods

4.78.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.78.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.78.5 Properties

4.78.6 assetTrack as AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The asset track for which the player item represents presentation state. (read-only)
Notes: (Read only property)

4.78.7 currentVideoFrameRate as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: For video media types, indicates the current frame rate of the track as it plays. (read-only)
Notes: If the media type of the assetTrack is AVMediaTypeVideo, the property indicates the current frame rate of the track as it plays, in units of frames per second.

If the item is not playing, or if the media type of the track is not video, the value of this property is 0.0.

This property is not key-value observable.
Available in OS X v10.9 and later.
(Read only property)

4.78.8 Enabled as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the track is enabled for presentation during playback.
Notes: (Read and Write property)

4.78.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.79 class AVPlayerItemVideoOutputMBS

4.79.1 class AVPlayerItemVideoOutputMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The AVPlayerItemVideoOutput lets you coordinate the output of content associated with a Core Video pixel buffer.
Notes: Available in OS X v10.8 and later.
This class can call the outputSequenceWasFlushed and outputMediaDataWillChange events in the AVFoundationMBS class.
Subclass of the AVPlayerItemOutputMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.5pr4

4.79.2 Methods

4.79.3 Constructor(pixelBufferAttributes as dictionary)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes and returns a video output object using the specified pixel buffer attributes.
Notes: pixelBufferAttributes; The pixel buffer attributes required for video output. For a list of pixel buffer attributes you can include in this dictionary, see the CVPixelBuffer.h header file in the Core Video framework.

4.79.4 copyCIImageForItemTime(time as CMTimeMBS) as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Acquires and returns an image that is appropriate to display at the specified time.
Notes: itemTime: The time at which you want to retrieve the image from the item.
outItemTimeForDisplay: Optional, the time by which you intend to use the returned pixel buffer.

Returns a pixel buffer containing the image data to display or nil if nothing should be displayed at the specified time. The caller is responsible for calling CVBufferRelease on the returned data when it is no longer needed.

Typically, you call this method in response to a CVDisplayLink callback or a CADisplayLink delegate method call when the hasNewPixelBufferForItemTime method also returns true.

After calling this method, the video output object marks the pixel buffer data as having been acquired. This causes the hasNewPixelBufferForItemTime method to return false unless newer data becomes available.

Returns CIImageMBS object.
Available in OS X v10.8 and later.
See also:

- 4.79.5 copyCIImageForItemTime(time as CMTimeMBS, byref outItemTimeForDisplay as CMTimeMBS) as Variant 587

4.79.5 copyCIImageForItemTime(time as CMTimeMBS, byref outItemTimeForDisplay as CMTimeMBS) as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Acquires and returns an image that is appropriate to display at the specified time.
Notes: itemTime: The time at which you want to retrieve the image from the item.
outItemTimeForDisplay: Optional, the time by which you intend to use the returned pixel buffer.

Returns a pixel buffer containing the image data to display or nil if nothing should be displayed at the specified time. The caller is responsible for calling CVBufferRelease on the returned data when it is no longer needed.

Typically, you call this method in response to a CVDisplayLink callback or a CADisplayLink delegate method call when the hasNewPixelBufferForItemTime method also returns true.

After calling this method, the video output object marks the pixel buffer data as having been acquired. This causes the hasNewPixelBufferForItemTime method to return false unless newer data becomes available.

Returns CIImageMBS object.
Available in OS X v10.8 and later.
See also:

- 4.79.4 copyCIImageForItemTime(time as CMTimeMBS) as Variant 586

4.79.6 copyPixelBufferForItemTime(time as CMTimeMBS) as CVPixelBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Acquires and returns an image that is appropriate to display at the specified time.
Notes: itemTime: The time at which you want to retrieve the image from the item.
outItemTimeForDisplay: Optional, the time by which you intend to use the returned pixel buffer.

Returns a pixel buffer containing the image data to display or nil if nothing should be displayed at the specified time. The caller is responsible for calling CVBufferRelease on the returned data when it is no longer needed.

Typically, you call this method in response to a CVDisplayLink callback or a CADisplayLink delegate method call when the hasNewPixelBufferForItemTime method also returns true.

After calling this method, the video output object marks the pixel buffer data as having been acquired. This causes the hasNewPixelBufferForItemTime method to return false unless newer data becomes available.

Returns CVPixelBufferMBS object.
Available in OS X v10.8 and later.
See also:

- 4.79.7 copyPixelBufferForItemTime(time as CMTimeMBS, byref outItemTimeForDisplay as CMTimeMBS) as CVPixelBufferMBS 588

4.79.7 copyPixelBufferForItemTime(time as CMTimeMBS, byref outItemTimeForDisplay as CMTimeMBS) as CVPixelBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Acquires and returns an image that is appropriate to display at the specified time.
Notes: itemTime: The time at which you want to retrieve the image from the item.
outItemTimeForDisplay: Optional, the time by which you intend to use the returned pixel buffer.

Returns a pixel buffer containing the image data to display or nil if nothing should be displayed at the specified time. The caller is responsible for calling CVBufferRelease on the returned data when it is no longer needed.

Typically, you call this method in response to a CVDisplayLink callback or a CADisplayLink delegate method call when the hasNewPixelBufferForItemTime method also returns true.

After calling this method, the video output object marks the pixel buffer data as having been acquired. This causes the hasNewPixelBufferForItemTime method to return false unless newer data becomes available.

Available in OS X v10.8 and later.
See also:

- 4.79.6 copyPixelBufferForItemTime(time as CMTimeMBS) as CVPixelBufferMBS 587

4.79.8 hasNewPixelBufferForItemTime(time as CMTimeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether video output is available for the specified item time.
Notes: itemTime: The item time to query. The time value is relative to the AVPlayerItem object with which the receiver is associated.

Returns true if there is available video output that has not been previously acquired or false if there is not.

This method returns true if the video data at the specified time has not yet been acquired or is different from the video that was acquired previously. If you require multiple objects to acquire video output from the same AVPlayerItem object, you should create separate AVPlayerItemVideoOutput objects for each.
Available in OS X v10.8 and later.

4.79.9 requestNotificationOfMediaDataChangeWithAdvanceInterval(time as Double)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Informs the receiver that the video out put client is entering a quiescent state.
Notes: interval: The amount of time to wait before notifying the delegate of the media change.

Call this method before you suspend your use of a CVDisplayLinkRef type or a CADisplayLink object. After the interval expires, the video output object notifies its delegate that it should resume the display link. If the interval value you specify is large, the delegate is notified as soon as possible rather than waiting.

Do not call this method repeatedly to force the delegate to be notified for each sample.

4.79.10 setDelegate

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sets the delegate so the events are called for this instance.
Notes: The constructor (if used) sets this for you automatically.

4.80 class AVPlayerLayerMBS

4.80.1 class AVPlayerLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVPlayerLayer is a subclass of CALayer to which an AVPlayer object can direct its visual output.
Notes: The videoGravity property defines how the video content is displayed within the player layer's bounds rect.

The value for the contents key of a player layer is opaque and effectively read-only.

During playback, AVPlayer may compensate for temporal drift between its visual output and its audible output to one or more independently-clocked audio output devices by adjusting the timing of its associated player layers. The effects of these adjustments are usually very small; however, clients that wish to remain entirely unaffected by such adjustments may wish to place other layers for which timing is important into independently timed subtrees of their layer trees.

You can create arbitrary numbers of player layers with the same AVPlayer object. Only the most-recently-created player layer will actually display the video content on-screen.
Subclass of the CALayerMBS class.

4.80.2 Methods

4.80.3 Constructor(player as AVPlayerMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a player layer to display the visual output of a specified player.
Notes: player: The player for which the player layer displays visual output.

4.80.4 isReadyForDisplay as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the first video frame has been made ready for display for the current item of the associated player. (read-only)
Notes: Use this property as an indicator of when best to show or animate-in a player layer into view. An player layer may be displayed, or made visible, while this property is false, however the layer will not have any user-visible content until the value becomes true.

This property remains false for a player's currentItem whose asset contains no enabled video tracks.

4.80.5 playerLayerWithPlayer(player as AVPlayerMBS) as AVPlayerLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a player layer to display the visual output of a specified player.
Notes: player: The player for which the player layer displays visual output.

Returns a player layer configured to display the visual output of player.

4.80.6 videoRect as CGRectMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The current size and position of the video image as displayed within the receiver's bounds. (read-only)
Notes: Available in OS X v10.9 and later.

4.80.7 Properties

4.80.8 player as AVPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The player for which the player layer displays visual output.
Notes: (Read and Write computed property)

4.80.9 videoGravity as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Specifies how the video is displayed within a player layer's bounds.
Notes: Options are AVLayerVideoGravityResizeAspect, AVLayerVideoGravityResizeAspectFill, and AVLayerVideoGravityResize.
This property is animatable.
(Read and Write computed property)

4.81 class AVPlayerLooperMBS

4.81.1 class AVPlayerLooperMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: AVPlayerLooperMBS is a helper object that repeatedly plays an AVPlayerItemMBS with an AVQueuePlayerMBS.
Example:

```
// Create player and configure
Dim player As New AVQueuePlayerMBS
Dim playerLayer As New AVPlayerLayerMBS(player)
player.pause

// Create looping item
Dim file As FolderItem = SpecialFolder.Desktop.Child("loop.mov")
Dim itemToLoop As New AVPlayerItemMBS(file)

// Create looping helper object. Loop item segment from 5sec to 7sec
Dim startTime As CMTimeMBS = CMTimeMBS.Make(5000, 1000)
Dim duration As CMTimeMBS = CMTimeMBS.Make(2000, 1000)
Dim timeRange As CMTimeRangeMBS = CMTimeRangeMBS.Make(startTime, duration)
Dim looper As New AVPlayerLooperMBS(player, itemToLoop, timeRange)

// Perform any other set up operations like setting AVPlayerItemDataOutputs on the looping item replicas
// Start playback
player.play

// itemToLoop between 5s and 7s plays repeatedly
'....
// To end the looping
looper.disableLooping
// Player will play through the end of the current looping item
```

Notes: The same result can be accomplished with AVQueuePlayer directly, but AVPlayerLooper provides a simpler interface to loop a single AVPlayerItem with an option to specify a time range. AVPlayerLooper only supports looping for forward playback (positive player rate). Behavior is undefined for negative player rate.

Note that the transition at the loop point is not guaranteed to be gapless.

Requires macOS 10.13, iOS 10.0 or newer.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3

- MBS Xojo Plugins in version 22.3
- MBS Xojo Plugins, version 22.3pr1

Xojo Developer Magazine

- 21.1, page 33: News from MBS Xojo Plugins, What's up with MonkeyBread Software by Stefanie Juchmes

4.81.2 Methods

4.81.3 available as boolean

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Whether the class is available.
Notes: Returns true on macOS 10.12 or newer as well as on iOS 10.0 or newer.

4.81.4 Constructor(player as AVQueuePlayerMBS, item as AVPlayerItemMBS)

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Returns an instance of AVPlayerLooper to loop specified AVPlayerItem with specified AVQueuePlayerMBS.
Notes: Equivalent to other constructor and passing in kCMTimeRangeInvalid for timeRange parameter.
See also:

- 4.81.5 Constructor(player as AVQueuePlayerMBS, item as AVPlayerItemMBS, loopRange as CMTimeRangeMBS) 593

4.81.5 Constructor(player as AVQueuePlayerMBS, item as AVPlayerItemMBS, loopRange as CMTimeRangeMBS)

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Returns an instance of AVPlayerLooper to loop specified AVPlayerItem within the specified time range with specified AVQueuePlayerMBS.
Notes: loopRange: Playback time range in [0, itemToLoop's duration] . kCMTimeRangeInvalid means [0, itemToLoop's duration] .

The specified AVPlayerItem will be used as a template to generate at least 3 AVPlayerItem replicas and the replicas will be inserted into specified AVQueuePlayer's play queue to accomplish the looping playback. The specified AVPlayerItem should have its asset's duration property loaded beforehand so looping setup work would not be blocked until the duration value is known. Otherwise, AVPlayerLooper's status property is

AVPlayerLooperStatusUnknown until the duration property is loaded. The specified AVPlayerItem will not be used in the actual looping playback. Furthermore, AVPlayerItem replicas will be generated at initialization time so any changes made to the specified AVPlayerItem's property afterwards will not be reflected in the replicas used for looping playback. Specified CMTimeRange will limit each item loop iteration to playing within the specified time range. To play from beginning and the whole duration of the item, specify kCMTimeRangeInvalid for the range parameter. Time range will be accomplished by seeking to range start time and setting AVPlayerItem's forwardPlaybackEndTime property on the looping item replicas. Client should not modify AVQueuePlayer's play queue while AVPlayerLooper is performing the looping. AVPlayerLooper will insert the replica items before any existing items in the specified AVQueuePlayer's play queue and change the actionAtItemEnd to AVPlayerActionAtItemEndAdvance if required. AVQueuePlayer's play queue and actionAtItemEnd will be restored when disableLooping method is called and then current looping item replicas completes playback or when AVPlayerLooper is destroyed. While AVPlayerLooper is being initialized, the specified AVQueuePlayer will be paused (rate of 0.0) if necessary and the original player rate will be restored after initialization completes. The client shall set the specified AVQueuePlayer's rate to 0 beforehand if additional set-up work needs to be performed after AVPlayerLooper initialization and before starting looping playback. An NSInvalidArgumentException will be raised if the player and template item are not specified or the template item has a 0 duration. An NSInvalidArgumentException will be raised if a valid time range has a duration of 0 or is not contained within time 0 and duration of the templateItem.
See also:

- 4.81.4 Constructor(player as AVQueuePlayerMBS, item as AVPlayerItemMBS) 593

4.81.6 disableLooping

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Disables the item looping.
Notes: AVPlayerLooper will stop performing player queue operations for looping and let the current looping item replica play to the end. The player's original actionAtItemEnd property will be restored afterwards. After this method is called, the value of the receiver's status property will be AVPlayerLooperStatusCancelled.

4.81.7 loopingPlayerItems as AVPlayerItemMBS()

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Returns an array containing replicas of specified AVPlayerItem used to accomplish the looping.
Notes: AVPlayerLooper creates replicas of the template AVPlayerItem using -copyWithZone: and inserts the replicas in the specified AVQueuePlayer to accomplish the looping. The AVPlayerItem replicas are for informational purposes and to allow the client to apply properties that are not transferred from the template AVPlayerItem to the replicas. The client can determine the number of replicas created and can listen for notifications and property changes from the replicas if desired. AVPlayerItemOutputs and AVPlayerItemMediaDataCollectors are not transferred to the replicas so the client should add them to each replica if desired. The client shall not modify the properties on the replicas that would disrupt looping playback. Examples of such properties are playhead time/date, selected media option, and forward playback end time.

This property is key value observable.

Returns an array containing replicas of specified AVPlayerItem.

4.81.8 Properties

4.81.9 error as NSErrorMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: If the receiver's status is AVPlayerLooperStatusFailed, this describes the error that caused the failure.
Notes: The value of this property is a NSError that describes what caused the receiver to not be able to perform looping playback. If the receiver's status is not AVPlayerLooperStatusFailed, the value of this property is nil.
(Read only property)

4.81.10 Handle as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

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

4.81.11 item as AVPlayerItemMBS

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: The item used.
Notes: (Read only property)

4.81.12 loopCount as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: Number of times the specified AVPlayerItem has been played.
Notes: Starts at 0 and increments when the player starts playback of the AVPlayerItem again.
(Read only property)

4.81.13 status as Integer

Plugin Version: 22.3, Platforms: macOS, iOS, Targets: All.

Function: The ability of the receiver to be used for looping playback.
Notes: The value of this property is an AVPlayerLooperStatus that indicates whether the receiver is ready for looping playback. When the value of this property is AVPlayerStatusFailed, the receiver can no longer be used for playback and a new instance needs to be created in its place. When this happens, clients can check the value of the error property to determine the nature of the failure. This property is key value observable. (Read only property)

4.81.14 Constants

Status

Constant	Value	Description
StatusCancelled	3	Indicates that the looper is no longer looping because -disableLooping was invoked.
StatusFailed	2	Indicates that the looper is not able to perform looping playback because of an error. The error is described by the value of the error property.
StatusReady	1	Indicates that the looper is ready for looping playback.
StatusUnknown	0	Indicates that the status of the looper is not yet known.

4.82 class AVPlayerMBS

4.82.1 class AVPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVPlayer object to implement controllers and user interfaces for single- or multiple-item playback.
Notes: The multiple-item case supports advanced behaviors.

AVPlayer works equally well with local and remote media files, providing you with appropriate information about readiness to play or about the need to await additional data before continuing.

You can display the visual content of items played by an instance of AVPlayer in a CoreAnimation layer of class AVPlayerLayer; to synchronize real-time playback with other CoreAnimation layers, you can use AVSynchronizedLayer. You cannot use an instance of AVVideoCompositionCoreAnimationTool with an AVPlayer object; for offline rendering you should instead use AVAssetExportSession.

You can observe the status of a player using key-value observing. So that you can add and remove observers safely, AVPlayer serializes notifications of changes that occur dynamically during playback on a dispatch queue. By default, this queue is the main queue . To ensure safe access to a player's nonatomic properties while dynamic changes in playback state may be reported, you must serialize access with the receiver's notification queue. In the common case, such serialization is naturally achieved by invoking AVPlayer's various methods on the main thread or queue.
Blog Entries

- New in the MBS Xojo Plugins Version 20.2
- MBS Xojo Plugins, version 19.0pr6
- MBS Xojo Plugins, version 18.4pr5
- Tip of the day: MoviePlayer.rate

Videos

- XDC 2020 MBS Plugins Presentation

Xojo Developer Magazine

- 18.3, page 56: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

4.82.2 Methods

4.82.3 addBoundaryTimeObserverForTime(time as CMTimeMBS, tag as variant = nil) as AVPlayerTimeObserverMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Requests invocation of a event when specified time are traversed during normal playback.
Notes: time: A CMTimeMBS value representing the times at which to invoke event.
Returns an opaque object that you pass as the argument to removeTimeObserver to stop observation.

You must retain the returned value as long as you want the time observer to be invoked by the player. Each invocation of this method should be paired with a corresponding call to removeTimeObserver. This is done automatically for you by AVPlayerTimeObserverMBS class destructor.

Calls AVFoundationMBS.BoundaryTimeObserver event passing tag value.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.82.4 addBoundaryTimeObserverForTimes(times() as CMTimeMBS, tag as Variant = nil) as AVPlayerTimeObserverMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Requests invocation of a event when specified times are traversed during normal playback.
Notes: times: An array of NSValue objects containing CMTime values representing the times at which to invoke event.
Returns an opaque object that you pass as the argument to removeTimeObserver to stop observation.

You must retain the returned value as long as you want the time observer to be invoked by the player. Each invocation of this method should be paired with a corresponding call to removeTimeObserver. This is done automatically for you by AVPlayerTimeObserverMBS class destructor.

Calls AVFoundationMBS.BoundaryTimeObserver event passing tag value.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.82.5 addLoopTimeObserver(atTime as CMTimeMBS, jumpToTime as CMTimeMBS) as AVPlayerTimeObserverMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Requests invocation of a code to jump in time to another position.
Notes: atTime: A CMTimeMBS value representing the times at which to invoke event.
jumpToTime: The CMTImeMBS where to jump to.

Returns an opaque object that you pass as the argument to removeTimeObserver to stop observation.

You must retain the returned value as long as you want the time observer to be invoked by the player. Each invocation of this method should be paired with a corresponding call to removeTimeObserver. This is done automatically for you by AVPlayerTimeObserverMBS class destructor.

4.82.6 addPeriodicTimeObserverForInterval(interval as CMTimeMBS, tag as Variant = nil) as AVPlayerTimeObserverMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Requests invocation of a given event during playback to report changing time.
Notes: interval: The interval of invocation of the event during normal playback, according to progress of the current time of the player.
The event takes a single parameter:

Returns an opaque object that you pass as the argument to removeTimeObserver to cancel observation.

You must retain the returned value as long as you want the time observer to be invoked by the player. Each invocation of this method should be paired with a corresponding call to removeTimeObserver. This is done automatically for you by AVPlayerTimeObserverMBS class destructor.

The AVFoundationMBS.PeriodicTimeObserver event is invoked periodically at the interval specified, interpreted according to the timeline of the current item. The event is also invoked whenever time jumps and whenever playback starts or stops. If the interval corresponds to a very short interval in real time, the player may invoke the event less frequently than requested. Even so, the player will invoke the event sufficiently often for the client to update indications of the current time appropriately in its end-user interface.

Special Considerations
Releasing the observer object without invoking removeTimeObserver: will result in undefined behavior.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.82.7 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.82.8 cancelPendingPrerolls

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels the preloading of media data.
Notes: This method cancels any pending operations to prepare the render pipeline for the current item.
Available in OS X v10.8 and later.

4.82.9 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.
See also:

- 4.82.10 Constructor(File as folderitem) 600
- 4.82.11 Constructor(item as AVPlayerItemMBS) 601
- 4.82.12 Constructor(URL as string) 601

4.82.10 Constructor(File as folderitem)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a new player to play a single audiovisual resource referenced by a given file.
Notes: File: A folderitem that identifies an audiovisual resource.
This method implicitly creates an AVPlayerItem object. You can get the player item using currentItem.
See also:

- 4.82.9 Constructor 600
- 4.82.11 Constructor(item as AVPlayerItemMBS) 601
- 4.82.12 Constructor(URL as string) 601

4.82.11 Constructor(item as AVPlayerItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a new player to play a given single audiovisual item.
Notes: You can use this method to play items for which you have an existing AVAsset object (see Constructor in AVPlayerItem).
See also:

- 4.82.9 Constructor 600
- 4.82.10 Constructor(File as folderitem) 600
- 4.82.12 Constructor(URL as string) 601

4.82.12 Constructor(URL as string)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes a new player to play a single audiovisual resource referenced by a given URL.
Notes: URL: An URL that identifies an audiovisual resource.
This method implicitly creates an AVPlayerItem object. You can get the player item using currentItem.
See also:

- 4.82.9 Constructor 600
- 4.82.10 Constructor(File as folderitem) 600
- 4.82.11 Constructor(item as AVPlayerItemMBS) 601

4.82.13 mediaSelectionCriteriaForMediaCharacteristic(mediaCharacteristic as string) as AVPlayerMediaSelectionCriteriaMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the automatic selection criteria for media that has the specified media characteristic.
Notes: mediaCharacteristic: The media characteristic for which the selection criteria is to be returned. Supported values include AVMediaCharacteristicAudible, AVMediaCharacteristicLegible, and AVMediaCharacteristicVisual.

Returns the automatic media selection criteria for mediaCharacteristic.

Available in OS X v10.9 and later.

4.82.14 pause

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Pauses playback.
Notes: This is the same as setting rate to 0.0.

4.82.15 play

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Begins playback of the current item.
Notes: This is the same as setting rate to 1.0.

4.82.16 playerWithFile(File as folderitem) as AVPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player to play a single audiovisual resource referenced by a given file.
Notes: File: A folderitem that identifies an audiovisual resource.
Returns a new player initialized to play the audiovisual resource specified by folderitem.
This method implicitly creates an AVPlayerItem object. You can get the player item using currentItem.

4.82.17 playerWithPlayerItem(item as AVPlayerItemMBS) as AVPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player initialized to play a given single audiovisual item
Notes: item: A player item.
Returns a new player, initialized to play item.

You can use this method to play items for which an AVAsset object has previously been created (see Constructor in AVPlayerItem).

4.82.18 playerWithURL(URL as string) as AVPlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new player to play a single audiovisual resource referenced by a given URL.
Notes: URL: An URL that identifies an audiovisual resource.
Returns a new player initialized to play the audiovisual resource specified by URL.

This method implicitly creates an AVPlayerItem object. You can get the player item using currentItem.

4.82.19 prerollAtRate(rate as Double, tag as Variant)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Begins loading media data to prime the media pipelines for playback.
Notes: rate: The playback rate to use when determining how much data to load.

Calls AVFoundationMBS.prerollAtRateFinished event when the player finishes the load attempt.

This method loads data starting at the item's current playback time. The current rate for the playback item should always be 0 prior to calling this method. After the method calls the completion handler, you can change the item's playback rate to begin playback.

If the player object is not ready to play (its status property is not AVPlayerStatusReadyToPlay), this method throws an exception.
Available in OS X v10.8 and later.

4.82.20 removeTimeObserver(observer as AVPlayerTimeObserverMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels a previously registered time observer.
Notes: observer: An object returned by a previous call to addPeriodicTimeObserverForInterval or addBoundaryTimeObserverForTimes.

Upon return, the caller is guaranteed that no new time observer event will begin executing. Depending on the calling thread and the queue used to add the time observer, an in-flight block may continue to execute after this method returns. You can guarantee synchronous time observer removal by enqueuing the call to removeTimeObserver on the main queue (do it in main event).

You should use this method to explicitly cancel each time observer added using addPeriodicTimeObserverForInterval and addBoundaryTimeObserverForTimes.

4.82.21 replaceCurrentItemWithPlayerItem(item as AVPlayerItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Replaces the player item with a new player item.

Notes: item: A player item.

You can only use this method with players created without queues. If the player was not initialized with a single item and no queue, the method throws an exception.

The item replacement occurs asynchronously; observe the currentItem property to find out when the replacement will/did occur.

Special Considerations
The new item must have the same compositor as the item it replaces, or have no compositor.

4.82.22 seekToDate(date as date, fireEvent as boolean = false, tag as Variant = nil)

Plugin Version: 14.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Moves the playback cursor to the specified time.
Notes: If FireEvent is true, the AVFoundationMBS.playerSeekToDateFinished event is called.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.82.23 seekToDate(date as dateTime, fireEvent as boolean = false, tag as variant = nil) 604

4.82.23 seekToDate(date as dateTime, fireEvent as boolean = false, tag as variant = nil)

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Moves the playback cursor to the specified time.
Notes: If FireEvent is true, the AVFoundationMBS.playerSeekToDateFinished event is called.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.82.22 seekToDate(date as date, fireEvent as boolean = false, tag as Variant = nil) 604

4.82.24 seekToTime(time as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Moves the playback cursor to a given time.
Notes: time: The time to which to move the playback cursor.

Use this method to seek to a specified time for the current player item and be notified when the operation completes. If the seek request completes without being interrupted (either by another seek request or by any other operation), the AVFoundationMBS.playerSeekToTimeFinished is executed with the finished parameter set to true.

If another seek request is already in progress when you call this method, the event for the in-progress seek request is executed immediately with the finished parameter set to false.

If fireevent is true, the AVFoundationMBS.playerSeekToTimeFinished event is run later.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.82.25 seekToTime(time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil) 605

4.82.25 seekToTime(time as CMTimeMBS, toleranceBefore as CMTimeMBS, toleranceAfter as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Moves the playback cursor within a specified time bound and invokes the event when the seek operation has either been completed or been interrupted.
Notes: If fireevent is true, the AVFoundationMBS.playerSeekToTimeFinished event is run later.

time: The time to which you would like to move the playback cursor.
toleranceBefore: The tolerance allowed before time.
toleranceAfter: The tolerance allowed after time.

Use this method to seek to a specified time for the current player item and to be notified when the seek operation is complete.

The time seeked to will be within the range [time-beforeTolerance, time+afterTolerance] , and may differ from the specified time for efficiency. If you pass kCMTimeZero for both toleranceBefore and toleranceAfter (to request sample accurate seeking), you may incur additional decoding delay.

Invoking this method with toleranceBefore set to kCMTimePositiveInfinity and toleranceAfter set to kCMTimePositiveInfinity is the same as invoking seekToTime:.

The completion handler for any prior seek request that is still in process will be invoked immediately with the finished parameter set to false. If the new request completes without being interrupted by another seek request or by any other operation the event will be invoked with the finished parameter set to true.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.
See also:

- 4.82.24 seekToTime(time as CMTimeMBS, fireEvent as boolean = false, tag as Variant = nil) 605

4.82.26 setMediaSelectionCriteria(criteria as AVPlayerMediaSelectionCriteriaMBS, mediaCharacteristic as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Applies automatic selection criteria for media that has the specified media characteristic.
Notes: criteria: An instance of AVPlayerMediaSelectionCriteria that specifies the selection criteria.
mediaCharacteristic: The media characteristic for which the selection criteria are to be applied. Supported values include AVMediaCharacteristicAudible, AVMediaCharacteristicLegible, and AVMediaCharacteristicVisual.

Criteria will be applied to an AVPlayerItem instance when:

- It is made ready to play.
- Specific media selections are made by the AVPlayerItem instance using the method selectMediaOption in a different group. The automatic choice in one group may be influenced by a specific selection in another group.
- Underlying system preferences change, e.g. system language, accessibility captions.

Specific selections made by the AVPlayerItem instance using the method selectMediaOption method within any group will override automatic selection in that group until the player item receives a selectMediaOptionAutomaticallyInMediaSelectionGroup message.

Available in OS X v10.9 and later.

4.82.27 setRate(rate as Double, time as CMTimeMBS, HostTime as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Synchronizes the playback rate and time of the current item with an external source.
Notes: rate: The playback rate for the item.
itemTime: The precise time at which to match playback of the item. To use the current item's current time, specify kCMTimeInvalid.
hostClockTime: The host time at which to synchronize playback. If you specify kCMTimeInvalid, the rate and time are set together without any external synchronization.

This method adjusts the current item's timebase so that the time in itemTime is in sync with the time in hostClockTime. Thus, if hostClockTime specifies a time in the past, the item's timebase is adjusted to make it appear as if the item has been running at the specified rate since itemTime. And if hostClockTime specifies a time in the future, playback is adjusted backward (if possible) so that the value in itemTime occurs at the precise moment the host's clock reaches the value in hostClockTime. If there is no content to play before the time specified by itemTime, playback holds until the two times come into sync.

This method does not ensure that media data is loaded before the timebase starts moving. However, if you specify a host time in the near future, that would give you some time to load the media data and prepare for playback.

4.82.28 Properties

4.82.29 ActionAtItemEnd as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The action to perform when an item has finished playing.
Notes: (Read and Write property)

4.82.30 appliesMediaSelectionCriteriaAutomatically as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Indicates whether the receiver should apply the current selection criteria automatically to AVPlayerItems.
Notes: By default, AVPlayer applies selection criteria based on system preferences. To override the default criteria for any media selection group, use setMediaSelectionCriteria method.

(Read and Write property)

4.82.31 audioOutputDeviceUniqueID as String

Plugin Version: 14.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Specifies the unique ID of the Core Audio output device used to play audio.
Notes: By default, the value of this property is nil, indicating that the default audio output device is used. Otherwise the value of this property is a string containing the unique ID of the Core Audio output device to be used for audio output.
Core Audio's kAudioDevicePropertyDeviceUID is a suitable source of audio output device unique IDs.
(Read and Write property)

4.82.32 ClosedCaptionDisplayEnabled as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the player uses closed captioning.
Notes: (Read and Write property)

4.82.33 currentItem as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The player's current item. (read-only)
Notes: (Read only property)

4.82.34 currentTime as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the current time of the current item.
Notes: (Read only property)

4.82.35 error as NSErrorMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: If the receiver's status is AVPlayerStatusFailed, this describes the error that caused the failure. (read-only)

Notes: The value of this property is an error object that describes what caused the receiver to no longer be able to play items. If the receiver's status is not AVPlayerStatusFailed, the value of this property is nil.
(Read only property)

4.82.36 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.82.37 Muted as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the audio output of the player is muted.
Notes: (Read and Write property)

4.82.38 Playing as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Whether asset is playing.
Notes: Returns true when playing.
(Read only property)

4.82.39 rate as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The current rate of playback.
Notes: 0.0 means "stopped", 1.0 means "play at the natural rate of the current item".
(Read and Write property)

4.82.40 status as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the player can be used for playback. (read-only)
Notes: When the value of this property is AVPlayerStatusFailed, you can no longer use the player for

playback and you need to create a new instance to replace it. If this happens, you can check the value of the error property to determine the nature of the failure.

This property is key value observable using key-value observing.
(Read only property)

4.82.41 volume as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the current audio volume of the player.
Notes: 0.0 means "silence all audio," 1.0 means "play at the full volume of the current item."
(Read and Write property)

4.82.42 Constants

Actions for the item end Constants

Constant	Value	Description
AVPlayerActionAtItemEndAdvance	0	Indicates that the player should advance to the next item, if there is one.
AVPlayerActionAtItemEndNone	2	Indicates that the player should do nothing.
AVPlayerActionAtItemEndPause	1	Indicates that the player should pause playing.

Status Constants

Constant	Value	Description
AVPlayerStatusFailed	2	Indicates that the player can no longer play AVPlayerItem instances because of an error. The error is described by the value of the player's error property.
AVPlayerStatusReadyToPlay	1	Indicates that the player is ready to play AVPlayerItem instances.
AVPlayerStatusUnknown	0	Indicates that the status of the player is not yet known because it has not tried to load new media resources for playback.

4.83 class AVPlayerMediaSelectionCriteriaMBS

4.83.1 class AVPlayerMediaSelectionCriteriaMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVPlayerMediaSelectionCriteria class specifies the preferred languages and media characteristics for an AVPlayer instance.
Notes: The languages and media characteristics of assets containing media selection options that an AVPlayer instance should attempt to select automatically when preparing and playing items. The languages and media characteristics are specified in the preferred order.

4.83.2 Methods

4.83.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.

4.83.4 Constructor(preferredLanguages() as string, preferredMediaCharacteristics() as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns an initialized media selection criteria instance with the specified settings.
Notes: preferredLanguages: An array of strings containing language identifiers, in the preferred order. Can be nil.
preferredMediaCharacteristics: An array of strings indicating additional media characteristics, the preferred order. Can be nil.
Supported media characteristics are defined in AVMediaSelectionOption Constants and Media Characteristics.

Returns an initialized AVPlayerMediaSelectionCriteria instance.

When making selections, AVPlayer treats the preferredLanguages as the paramount criterion and the preference for preferredMediaCharacteristics as secondary.

The objects in the preferredLanguages array are indicated using BCP 47 language identifiers or ISO 639-2/T language codes.

If no option with any of the preferredLanguages is available, a selection is made according to the default enabling and disabling of media options as stored in the asset.

The preferredMediaCharacteristics are used when selecting media for the AVPlayer.

For example, desirable characteristics of legible media may include AVMediaCharacteristicTranscribesSpokenDialogForAccessibility and AVMediaCharacteristicDescribesMusicAndSoundForAccessibility.

Similarly, desirable characteristics of audible media may include AVMediaCharacteristicDescribesVideoForAccessibility.

If no option is found that possesses all of the desired characteristics, the option that best matches the desired characteristics will be selected.
Available in OS X v10.9 and later.

4.83.5 preferredLanguages as String()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The array of preferred languages in the order of desirability. (read-only)
Notes: Available in OS X v10.9 and later.

4.83.6 preferredMediaCharacteristics as String()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The array of preferred media characteristics in the order of desirability. (read-only)
Notes: Available in OS X v10.9 and later.

4.83.7 Properties

4.83.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.84 class AVPlayerTimeObserverMBS

4.84.1 class AVPlayerTimeObserverMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The class for an observer.
Notes: You keep reference to this class, so you can cancel time observer later.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.84.2 Methods

4.84.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.84.4 Destructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The destructor.
Notes: Automatically cancels this observer if it was not cancelled.

4.85 control AVPlayerViewControlMBS

4.85.1 control AVPlayerViewControlMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The player control for Mac.
Notes: AVPlayerView is a subclass of NSView that can be used to display the visual content of an AVPlayer object and the standard playback controls.
MBS Plugin wraps this as a control for Xojo.
Please notice this is 64-bit only and requires MacOS 10.9 or newer.
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
- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr2

Videos

- Presentation from Munich conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

4.85.2 Methods

4.85.3 Available as Boolean

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

4.85.4 beginTrimming

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Sets the controls panel into trimming mode and blocks until the user selects either the Trim or the Cancel button.
Notes: Triggers TrimmingCompleted later.

4.85.5 flashChapterNumber(chapterNumber as Integer, chapterTitle as String)

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Display the provided chapter number and title momentarily.
Notes: chapterNumber: The chapter number (required).
chapterTitle: The chapter title (optional).

4.85.6 Properties

4.85.7 actionPopUpButtonMenu as NSMenuMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The action popup menu.
Notes: Clients can set this property in order to show an action pop up button. Default is nil.
(Read and Write property)

4.85.8 allowsPictureInPicturePlayback as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: Whether or not the receiver allows Picture in Picture playback.
Notes: Default is false.
Available in MacOS 10.15 or later.
(Read and Write property)

4.85.9 canBeginTrimming as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Whether or not the current media can be trimmed.
Notes: (Read only property)

4.85.10 contentOverlayView as NSViewMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Use the content overlay view to add additional custom views between the video content and the controls.

Notes: (Read only property)

4.85.11 controlsStyle as Integer

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The style of the playback controls pane currently associated with the view.
Notes: (Read and Write property)

4.85.12 player as AVPlayerMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The player from which to source the media content for the view.
Notes: (Read and Write property)

4.85.13 readyForDisplay as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Boolean indicating that the first video frame has been made ready for display for the current item of the associated AVPlayer.
Notes: (Read only property)

4.85.14 showsFrameSteppingButtons as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Replace scanning controls in the playback UI with frame stepping buttons.
Notes: Default is false.
(Read and Write property)

4.85.15 showsFullScreenToggleButton as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Whether or not the controls pane will show a full screen toggle button.
Notes: Default is false.
(Read and Write property)

4.85.16 showsSharingServiceButton as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Whether or not the controls pane will show a sharing service button when the current player item can be shared.
Notes: Default is false.
(Read and Write property)

4.85.17 showsTimecodes as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: Desktop only.

Function: If timecodes are available, allow the AVPlayerView controls to enter timecode mode.
Notes: Default is false.
Available in MacOS 10.15 or later.
(Read and Write property)

4.85.18 updatesNowPlayingInfoCenter as Boolean

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Whether or not the now playing info center should be updated.
Notes: Default is true.
(Read and Write property)

4.85.19 videoBounds as NSRectMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The current size and position of the video image as displayed within the receiver's view's bounds.
Notes: (Read only property)

4.85.20 videoGravity as String

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: A string defining how the video is displayed within an AVPlayerLayer bounds rect.
Notes: Options are AVLayerVideoGravityResizeAspect, AVLayerVideoGravityResizeAspectFill and AVLayerVideoGravityResize. AVLayerVideoGravityResizeAspect is default.
See <AVFoundation/AVAnimation.h>for a description of these options.

(Read and Write property)

4.85.21 View as NSViewMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: The reference to the AVPlayerView object.
Notes: (Read only property)

4.85.22 Events

4.85.23 BoundsChanged

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.85.24 Close

Plugin Version: 18.3, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

4.85.25 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.

4.85.26 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 18.3, 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.

4.85.27 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.

4.85.28 EnableMenuItems

Plugin Version: 18.3, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

4.85.29 FrameChanged

Plugin Version: 18.3, 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.

4.85.30 GotFocus

Plugin Version: 18.3, 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.

4.85.31 LostFocus

Plugin Version: 18.3, 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.

4.85.32 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 18.3, 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.

4.85.33 MouseDrag(x as Integer, y as Integer)

Plugin Version: 18.3, 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.

4.85.34 MouseUp(x As Integer, y As Integer)

Plugin Version: 18.3, 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.

4.85.35 Open

Plugin Version: 18.3, 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.

4.85.36 playerViewDidStartPictureInPicture

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture did start.

4.85.37 playerViewDidStopPictureInPicture

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture did stop.

4.85.38 playerViewFailedToStartPictureInPicture(error as NSErrorMBS)

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture failed to start.
Notes: error: An error describing why it failed.

4.85.39 playerViewRestoreUserInterfaceForPictureInPictureStop as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to restore the user interface before Picture in Picture stops.
Notes: Please return true if you restored it or false if not, so we can call completion handler passing the value.
Default is to pass true if you don't implement this event.

4.85.40 playerViewShouldAutomaticallyDismissAtPictureInPictureStart as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this event and return false to prevent player view from automatically being miniaturized or losing focus when Picture in Picture starts.
Notes: If you implement this, please return true or false.
By default we return false if you don't

4.85.41 playerViewWillStartPictureInPicture

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture will start.

4.85.42 playerViewWillStopPictureInPicture

Plugin Version: 20.2, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture will stop.

4.85.43 ScaleFactorChanged(NewFactor as double)

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

4.85.44 TrimmingCompleted(result as Integer)

Plugin Version: 18.3, Platform: macOS, Targets: .

Function: The event handler that is executed when the user selects either the Trim or Cancel button in the trimming UI.

4.85.45 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.

4.85.46 Constants

Control Styles

Constant	Value	Description
ControlsStyleDefault	1	The default controls pane is associated with the view.
ControlsStyleFloating	2	The floating controls pane is associated with the view.
ControlsStyleInline	1	The inline controls pane is associated with the view.
ControlsStyleMinimal	3	The minimal controls pane is associated with the view.
ControlsStyleNone	0	No controls pane is associated with the view.

Trim Results

Constant	Value	Description
TrimCancelButton	1	The user selected the Cancel button.
TrimOKButton	0	The user selected the Trim button.

4.86 class AVQueuePlayerMBS

4.86.1 class AVQueuePlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVQueuePlayer is a subclass of AVPlayer you use to play a number of items in sequence.
Notes: Subclass of the AVPlayerMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 22.3
- MBS Xojo Plugins, version 22.3pr1

4.86.2 Methods

4.86.3 advanceToNextItem

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Ends playback of the current item and initiates playback of the next item in the player's queue.
Notes: This method also removes the current item from the play queue.

4.86.4 appendItem(item as AVPlayerItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Places given player item in the queue.
Notes: item: The item to be inserted.

4.86.5 canAppendItem(item as AVPlayerItemMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given player item can be appended into the player's queue.
Notes: item: The AVPlayerItem object to test.

Returns true if item can be appended to the queue, otherwise false.
Adding the same item to a player at more than one position in the queue is not supported.

4.86.6 canInsertItem(item as AVPlayerItemMBS, afterItem as AVPlayerItemMBS = nil) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given player item can be inserted into the player's queue.
Notes: item: The AVPlayerItem object to test.
afterItem: The item that item is to follow in the queue. Pass nil to test whether item can be appended to the queue.

Returns true if item can be appended to the queue, otherwise false.
Adding the same item to a player at more than one position in the queue is not supported.

4.86.7 Constructor(items() as AVPlayerItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an instance of AVQueuePlayer by enqueueing the player items from a given array.
Notes: items: An array of AVPlayerItem objects with which initially to populate the player's queue.
Creates an instance of AVQueuePlayer initialized to play the player items in items.

4.86.8 insertItem(item as AVPlayerItemMBS, afterItem as AVPlayerItemMBS = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Places given player item after a specified item in the queue.
Notes: item: The item to be inserted.
afterItem: The item that the newly inserted item should follow in the queue. Pass nil to append the item to the queue.

4.86.9 items as AVPlayerItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the currently enqueued items.
Notes: The array contains AVPlayerItem objects.

4.86.10 queuePlayerWithItems(items() as AVPlayerItemMBS) as AVQueuePlayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an instance of AVQueuePlayer initialized to play items from a given array.
Notes: items: An array of AVPlayerItem objects with which initially to populate the player's queue.

Returns an instance of AVQueuePlayer initialized to play the player items in items.

4.86.11 removeAllItems

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes all the items from the queue.
Notes: This has the side-effect of stopping playback by the player.

4.86.12 removeItem(item as AVPlayerItemMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Removes a given player item from the queue.
Notes: item: The item to be removed.

If item is currently playing, this has the same effect as advanceToNextItem.

4.87 class AVRouteDetectorMBS

4.87.1 class AVRouteDetectorMBS

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: An object that detects the presences of media playback routes.
Notes: When route detection is enabled, AVRouteDetector reports whether multiple playback routes have been detected. If this is the case, AVRoutePickerViewControlMBS can be used to allow users to pick from the set of available routes. By default, route detection is disabled.

Available on MacOS 10.13 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 20.1pr2
- Airplay related classes for AVFoundation

Videos

- XDC 2020 MBS Plugins Presentation

Xojo Developer Magazine

- 18.3, page 56: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes
- 18.3, page 10: News

4.87.2 Methods

4.87.3 available as boolean

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: Whether to class is available.
Notes: Returns true on MacOS 10.13 or newer.

4.87.4 Constructor

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: The constructor.

4.87.5 Destructor

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: The destructor

4.87.6 Properties

4.87.7 Handle as Integer

Plugin Version: 20.1, Platform: macOS, Targets: All.

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

4.87.8 MultipleRoutesDetected as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether more playback routes than the local playback route have been detected.
Notes: When the value of this property changes MultipleRoutesDetectedDidChange event is called.
(Read only property)

4.87.9 RouteDetectionEnabled as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether route detection is enabled.
Notes: The default value is false.
Route detection significantly increases power consumption and must be turned off when it's no longer needed.
(Read and Write property)

4.87.10 Events

4.87.11 MultipleRoutesDetectedDidChange

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: The event called when MultipleRoutesDetected property changed.

4.88 control AVRoutePickerViewControlMBS

4.88.1 control AVRoutePickerViewControlMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: An object that displays controls for picking playback routes.
Notes: Available on MacOS 10.15 or newer.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- MBS Xojo Plugins, version 20.1pr2
- Airplay related classes for AVFoundation

Videos

- XDC 2020 MBS Plugins Presentation

Xojo Developer Magazine

- 18.3, page 10: News

4.88.2 Properties

4.88.3 player as AVPlayerMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: The player for which to perform routing operations.
Notes: (Read and Write property)

4.88.4 routePickerButtonBordered as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: Whether or not the picker button has a border.
Notes: Default is true.
(Read and Write property)

4.88.5 View as NSViewMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: The view object.
Notes: (Read only property)

4.88.6 Available as Boolean

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.
Notes: (Read and Write computed property)

4.88.7 routePickerButtonColorForState(state as integer) as NSColorMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop only.

Function: Defines the color for the button for a given state.
Notes: (Read and Write computed property)

4.88.8 Events

4.88.9 BoundsChanged

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.88.10 Close

Plugin Version: 20.1, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

4.88.11 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.

4.88.12 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 20.1, 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.

4.88.13 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.

4.88.14 DidEndPresentingRoutes

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: Tells you that the route picker view has finished presenting routes to the user.

4.88.15 EnableMenuItems

Plugin Version: 20.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

4.88.16 FrameChanged

Plugin Version: 20.1, 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.

4.88.17 GotFocus

Plugin Version: 20.1, 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.

4.88.18 LostFocus

Plugin Version: 20.1, 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.

4.88.19 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 20.1, 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.

4.88.20 MouseDrag(x as Integer, y as Integer)

Plugin Version: 20.1, 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.

4.88.21 MouseUp(x As Integer, y As Integer)

Plugin Version: 20.1, 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.

4.88.22 Open

Plugin Version: 20.1, 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.

4.88.23 ScaleFactorChanged(NewFactor as double)

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

4.88.24 WillBeginPresentingRoutes

Plugin Version: 20.1, Platform: macOS, Targets: .

Function: Tells you that the route picker view will start presenting routes to the user.

4.88.25 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.

4.88.26 Constants

Button States

Constant	Value	Description
StateActive	2	Active state of the picker. The picker has this state when AirPlay is active.
StateActiveHighlighted	3	Highlighted state of the active picker. The picker has this state when it is highlighted and AirPlay is active.
StateNormal	0	Normal or default state of the picker.
StateNormalHighlighted	1	Highlighted state of the picker. The picker has this state when a mouse-down event occurs inside the button. It loses this highlight when a mouse-up event occurs.

4.89 class AVSampleBufferDisplayLayerMBS

4.89.1 class AVSampleBufferDisplayLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVSampleBufferDisplayLayer is a subclass of CALayer that can decompress and display compressed or uncompressed video frames.
Notes: Subclass of the CALayerMBS class.

4.89.2 Methods

4.89.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.

4.89.4 enqueueSampleBuffer(sampleBuffer as CMSampleBufferMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Sends a sample buffer for display.
Notes: If sampleBuffer has the kCMSampleAttachmentKey_DoNotDisplay attachment set to True, the frame will be decoded but not displayed. Otherwise, if sampleBuffer has the kCMSampleAttachmentKey_DisplayImmediately attachment set to True, the decoded image will be displayed as soon as possible, replacing all previously enqueued images regardless of their timestamps. Otherwise, the decoded image will be displayed at sampleBuffer's output presentation timestamp, as interpreted by the control timebase (or the mach_absolute_time timeline if there is no control timebase). To schedule the removal of previous images at a specific timestamp, enqueue a marker sample buffer containing no samples, with the kCMSampleBufferAttachmentKey_EmptyMedia attachment set to kCFBooleanTrue. IMPORTANT NOTE: attachments with the kCMSampleAttachmentKey_ prefix must be set via CMSampleBufferGetSampleAttachmentsArray and CFDictionarySetValue. Attachments with the kCMSampleBufferAttachmentKey_ prefix must be set via CMSetAttachment.

4.89.5 flush

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the layer to discard pending enqueued sample buffers.
Notes: It is not possible to determine which sample buffers have been decoded, so the next frame passed

to enqueueSampleBuffer should be an IDR frame (also known as a key frame or sync sample).

4.89.6 flushAndRemoveImage

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the layer to discard pending enqueued sample buffers and remove any currently displayed image.
Notes: It is not possible to determine which sample buffers have been decoded, so the next frame passed to enqueueSampleBuffer should be an IDR frame (also known as a key frame or sync sample).

4.89.7 requestMediaDataWhenReady(tag as Variant = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Instructs the target to invoke a client-supplied block repeatedly, at its convenience, in order to gather sample buffers for display.
Notes: The block should enqueue sample buffers to the layer either until the layer's readyForMoreMediaData property becomes false or until there is no more data to supply. When the layer has decoded enough of the media data it has received that it becomes ready for more media data again, it will invoke the block again in order to obtain more. If this function is called multiple times, only the last call is effective.
Call stopRequestingMediaData to cancel this request.
Each call to requestMediaDataWhenReady should be paired
with a corresponding call to stopRequestingMediaData. Releasing the
AVSampleBufferDisplayLayer without a call to stopRequestingMediaData will result in undefined behavior.
Calls SampleBufferDisplayLayerMediaDataWhenReady event on AVFoundationMBS class.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.89.8 stopRequestingMediaData

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Cancels any current requestMediaDataWhenReady call.
Notes: This method may be called from outside the block or from within the block.

4.89.9 Properties

4.89.10 isReadyForMoreMediaData as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates the readiness of the layer to accept more sample buffers.
Notes: AVSampleBufferDisplayLayer keeps track of the occupancy levels of its internal queues for the benefit of clients that enqueue sample buffers from non-real-time sources – i.e., clients that can supply sample buffers faster than they are consumed, and so need to decide when to hold back. Clients enqueueing sample buffers from non-real-time sources may hold off from generating or obtaining more sample buffers to enqueue when the value of readyForMoreMediaData is false. It is safe to call enqueueSampleBuffer when readyForMoreMediaData is false, but it is a bad idea to enqueue sample buffers without bound. To help with control of the non-real-time supply of sample buffers, such clients can use requestMediaDataWhenReady in order to specify a block that the layer should invoke whenever it's ready for sample buffers to be appended.
The value of readyForMoreMediaData will often change from false to true asynchronously, as previously supplied sample buffers are decoded and displayed.
This property is not key value observable.
(Read only property)

4.89.11 videoGravity as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A string defining how the video is displayed within an AVSampleBufferDisplayLayer bounds rect.
Notes: Options are AVLayerVideoGravityResizeAspect, AVLayerVideoGravityResizeAspectFill and AVLayerVideoGravityResize. AVLayerVideoGravityResizeAspect is default.
See AVFoundationMBS class for a description of these options.
(Read and Write property)

4.90 class AVSynchronizedLayerMBS

4.90.1 class AVSynchronizedLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVSynchronizedLayer a subclass of CALayer with layer timing that synchronizes with a specific AVPlayerItem.
Notes: You can create an arbitrary number of synchronized layers from the same AVPlayerItem object.

A synchronized layer is similar to a CATransformLayer object in that it doesn't display anything itself, it just confers state upon its layer subtree. AVSynchronizedLayer confers is timing state, synchronizing the timing of layers in its subtree with that of a player item.
Subclass of the CALayerMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.90.2 Methods

4.90.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.90.4 synchronizedLayerWithPlayerItem(playerItem as AVPlayerItemMBS) as AVSynchronizedLayerMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new synchronized layer with timing synchronized with a given player item.
Notes: playerItem: A player item.

Returns a new synchronized layer with timing synchronized with playerItem.

4.90.5 Properties

4.90.6 playerItem as AVPlayerItemMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The player item to which the timing of the layer is synchronized.
Notes: (Read and Write computed property)

4.91 class AVTextStyleRuleMBS

4.91.1 class AVTextStyleRuleMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: AVTextStyleRule represents a set of text styling attributes that can be applied to some or all of the text of legible media, such as subtitles and closed captions..

4.91.2 Methods

4.91.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9.

4.91.4 Constructor(textMarkupAttributes as dictionary)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates an instance of AVTextStyleRule with the specified text markup attributes.
Notes: textMarkupAttributes: A dictionary with keys representing text style attributes that are specifiable in text markup. Eligible keys are defined in <CoreMedia/CMTextMarkup.h>.
See also:

- 4.91.5 Constructor(textMarkupAttributes as dictionary, textSelector as string) 641

4.91.5 Constructor(textMarkupAttributes as dictionary, textSelector as string)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates an instance of AVTextStyleRule with the specified text markup attributes and an identifier for the range or ranges of text to which the attributes should be applied.
Notes: textMarkupAttributes: A dictionary with keys representing text style attributes that are specifiable in text markup. Eligible keys are defined in <CoreMedia/CMTextMarkup.h>.
textSelector: An identifier for the range or ranges of text to which the attributes should be applied. Eligible identifiers are determined by the format and content of the legible media. A value of nil indicates that the textMarkupAttributes should be applied as default styles for all text unless overridden by content markup or other applicable text selectors.
Returns an instance of AVTextStyleRule
See also:

- 4.91.4 Constructor(textMarkupAttributes as dictionary) 641

4.91.6 copy as AVTextStyleRuleMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates a copy of the style rule.

4.91.7 textStyleRuleWithTextMarkupAttributes(textMarkupAttributes as Dictionary) as AVTextStyleRuleMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates an instance of AVTextStyleRule with the specified text markup attributes..
Notes: textMarkupAttributes: A dictionary with keys representing text style attributes that are specifiable in text markup. Eligible keys are defined in <CoreMedia/CMTextMarkup.h>.
Returns an instance of AVTextStyleRule
Equivalent to invoking textStyleRuleWithTextMarkupAttributes with a value of nil for textSelector.
See also:

- 4.91.8 textStyleRuleWithTextMarkupAttributes(textMarkupAttributes as Dictionary, textSelector as string) as AVTextStyleRuleMBS 642

4.91.8 textStyleRuleWithTextMarkupAttributes(textMarkupAttributes as Dictionary, textSelector as string) as AVTextStyleRuleMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates an instance of AVTextStyleRule with the specified text markup attributes and an identifier for the range or ranges of text to which the attributes should be applied.
Notes: textMarkupAttributes: A dictionary with keys representing text style attributes that are specifiable in text markup. Eligible keys are defined in <CoreMedia/CMTextMarkup.h>.
textSelector: An identifier for the range or ranges of text to which the attributes should be applied. Eligible identifiers are determined by the format and content of the legible media. A value of nil indicates that the textMarkupAttributes should be applied as default styles for all text unless overridden by content markup or other applicable text selectors.
Returns an instance of AVTextStyleRule
See also:

- 4.91.7 textStyleRuleWithTextMarkupAttributes(textMarkupAttributes as Dictionary) as AVTextStyleRuleMBS 642

4.91.9 Properties

4.91.10 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.91.11 textMarkupAttributes as Dictionary

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A dictionary with keys representing text style attributes that are specifiable in text markup.
Notes: Eligible keys and the expected types of their corresponding values are defined in <CoreMedia/CM-TextMarkup.h>.
(Read only property)

4.91.12 textSelector as String

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A string that identifies the range or ranges of text to which the attributes should be applied.
Notes: A value of nil indicates that the textMarkupAttributes should be applied as default styles for all text unless overridden by content markup or other applicable text selectors.
The syntax of text selectors is determined by the format of the legible media. Eligible selectors may be determined by the content of the legible media (e.g. CSS selectors that are valid for a specific WebVTT document).
(Read only property)

4.92 class AVTimeCodeMBS

4.92.1 class AVTimeCodeMBS

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The class for a time code.
Notes: The data class used by AVAssetMBS.readTimeCodeObjects to store details on time codes.
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 17.5
- MBS Xojo Plugins, version 17.5pr9

Videos

- Presentation from Munich conference about MBS Plugins.

Xojo Developer Magazine

- 16.1, page 9: News

4.92.2 Methods

4.92.3 Constructor

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: Private constructor.

4.92.4 Properties

4.92.5 duration as CMTimeMBS

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The duration.
Notes: (Read only property)

4.92.6 frameNumber as Int64

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The frame number.
Notes: (Read only property)

4.92.7 frameQuanta as UInt32

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: Returns the frames/sec for timecode (eg. 30) OR frames/tick for counter mode
Notes: (Read only property)

4.92.8 presentationTimeStamp as CMTimeMBS

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The earliest presentation timestamp.
Notes: (Read only property)

4.92.9 tcFlag as UInt32

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The time zone flags.
Notes: Can be combination of

kDropFrame	1
k24HourMax	2
kNegTimesOK	4

(Read only property)

4.92.10 timecode as String

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The timecode as formatted number.
Notes: Format is 2 digit hours, double colon, 2 digit minutes, double colon, 2 digit seconds, double colon, 2 digit frames.
e.g. "01:02:03:04"

(Read only property)

4.92.11 type as String

Plugin Version: 17.5, Platform: macOS, Targets: All.

Function: The type of timecode.
Notes: Can be tmcd for 32-bit time code or tc64 for 64-bit timecode.
(Read only property)

4.93 class AVTimedMetadataGroupMBS

4.93.1 class AVTimedMetadataGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVTimedMetadataGroup object to represent a collection of metadata items.
Notes: AV Foundation also provides a mutable subclass, AVMutableTimedMetadataGroup.

4.93.2 Methods

4.93.3 Constructor(items() as AVMetadataItemMBS, timeRange as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a metadata group initialized with given metadata items.
Notes: items: An array of AVMetadataItem objects.
timeRange: The time range of the metadata contained in items.

4.93.4 copy as AVTimedMetadataGroupMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.93.5 items as AVMetadataItemMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The metadata items in the group. (read-only)
Notes: The array contains instances of AVMetadataItem.

4.93.6 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range of the metadata. (read-only)

4.93.7 Properties

4.93.8 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.94 class AVURLAssetMBS

4.94.1 class AVURLAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: AVURLAsset is a concrete subclass of AVAsset that you use to initialize an asset from an URL..
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim u as AVURLAssetMBS = AVURLAssetMBS.URLAssetWithFile(f)
dim a() as AVAssetTrackMBS = u.tracks

MsgBox str(UBound(a)+1)+" tracks."
```

Notes: Subclass of the AVAssetMBS class.

4.94.2 Methods

4.94.3 audiovisualMIMETypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the MIME types the AVURLAsset class understands.
Example:

```
MsgBox Join(AVURLAssetMBS.audiovisualMIMETypes, EndOfLine)
```

Notes: See also MimeTypeToFileExtensionMBS function.

4.94.4 audiovisualTypes as string()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an array of the file types the AVURLAsset class understands.
Example:

```
MsgBox Join(AVURLAssetMBS.audiovisualTypes, EndOfLine)
```

4.94.5 compatibleTrackForCompositionTrack(compositionTrack as AVCompositionTrackMBS) as AVAssetTrackMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset track from which any time range can be inserted into a given composition track.
Notes: compositionTrack: The composition track for which a compatible AVAssetTrack object is requested.

Returns an asset track managed by the receiver from which any time range can be inserted into a given composition track.

You insert the track into using insertTimeRange (AVMutableCompositionTrack). This method is the logical complement of mutableTrackCompatibleWithTrack.

4.94.6 Constructor(File as folderitem, options as dictionary = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset for inspection of a resource referenced by a given file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim u as AVURLAssetMBS = new AVURLAssetMBS(f)
MsgBox str(U.duration.Seconds)+" seconds long"
```

Notes: file: A folderitem that references the container file to be represented by the asset.
options: A dictionary that contains options for the initialization of the asset.
See also:

- 4.94.7 Constructor(URL as string, options as dictionary = nil) 650

4.94.7 Constructor(URL as string, options as dictionary = nil)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes an asset for inspection of a resource referenced by a given URL.
Notes: URL: An URL that references the container file to be represented by the asset.
options: A dictionary that contains options for the initialization of the asset.
See also:

- 4.94.6 Constructor(File as folderitem, options as dictionary = nil) 650

4.94.8 isPlayableExtendedMIMEType(extendedMIMEType as string) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether an asset is playable with the codec(s) and container type specified in a given extended MIME type.
Notes: Returns true if an asset is playable with the codec(s) and container type specified in extendedMIMEType, otherwise false.
See also FileExtensionToMimeTypeMBS function.

4.94.9 URLAssetWithFile(File as folderitem, options as dictionary = nil) as AVURLAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset for inspection of a resource referenced by a given file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim u as AVURLAssetMBS = AVURLAssetMBS.URLAssetWithFile(f)

MsgBox str(u.duration.Seconds)+" seconds video."
```

Notes: File: A folderitem that references the container file to be represented by the asset.
options: A dictionary that contains options for the initialization of the asset.

Returns an asset initialized for inspection of a resource referenced by folderitem.

4.94.10 URLAssetWithURL(URL as string, options as dictionary = nil) as AVURLAssetMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns an asset for inspection of a resource referenced by a given URL.
Notes: URL: An URL that references the container file to be represented by the asset.
options: A dictionary that contains options for the initialization of the asset.

Returns an asset initialized for inspection of a resource referenced by URL.

4.94.11 Properties

4.94.12 resourceLoader as AVAssetResourceLoaderMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The resource loader associated with the asset. (read-only)
Notes: During loading, the resource loader object may be asked to assist in the loading of a resource. For example, a resource that requires decryption might result in the resource loader being asked to provide the appropriate decryption keys. You can assign a delegate object to the resource loader object and use your delegate to intercept these requests and provide an appropriate response.
Available on Mac OS X 10.9 or later.
(Read only property)

4.94.13 URL as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The URL with which the asset was initialized. (read-only)
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mov")
dim u as AVURLAssetMBS = new AVURLAssetMBS(f)
MsgBox u.URL
```

Notes: (Read only property)

4.95 class AVVideoCompositingMBS

4.95.1 class AVVideoCompositingMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVVideoCompositing protocol defines properties and methods that custom video compositors must implement.
Notes: For each AV Foundation object of class AVPlayerItem, AVAssetExportSession, AVAssetImageGenerator, or AVAssetReaderVideoCompositionOutput that has a non-nil value for its videoComposition property, and the value of the customVideoCompositorClass property of the AVVideoComposition is not nil, AV Foundation creates and uses an instance of that custom video compositor class to process the instructions contained in the AVVideoComposition.

The custom video compositor instance will be created when you assign videoComposition an instance of AVVideoComposition that's associated with a different custom video compositor class than the object was previously using.

When creating instances of custom video compositors, AV Foundation initializes them by calling init and then makes them available as the value of the customVideoCompositor property of the object to which it was assigned. You then can do any additional setup or configuration to the custom compositor.

Custom video compositor instances will then be retained by the AV Foundation object for as long as the value of its videoComposition property indicates that an instance of the same custom video compositor class should be used, even if the value is changed from one instance of AVVideoComposition to another instance that's associated with the same custom video compositor 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 / Real Studio Plugins, version 14.5pr3

4.95.2 Methods

4.95.3 cancelAllPendingVideoCompositionRequests

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Directs a custom video compositor object to cancel or finish all pending video composition requests.
Notes: Upon receiving this message, a custom video compositor must block until it has either cancelled all pending frame requests, and called the finishCancelledRequest method for each of them. If cancellation is not possible, the method must block until it has finished processing of all the frames and called the finishWithComposedVideoFrame method for each of them.
Available in OS X v10.9 and later.

4.95.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The private constructor.
Notes: Initializes an instance of the class that implements the video compositing protocol. (required)
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.

4.95.5 renderContextChanged(newRenderContext as AVVideoCompositionRenderContextMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Invoked to notify the custom compositor that a composition will switch to a different render context. (required)
Notes: newRenderContext: The new render context that will be handling video composition.

Instances of classes implementing the AVVideoComposting protocol must implement this method to be notified when the AVVideoCompositionRenderContext instance handing a video composition changes. AVVideoCompositionRenderContext instances being immutable, such a change will occur every time there is a change in the video composition parameters.
Available in OS X v10.9 and later.

4.95.6 startVideoCompositionRequest(asyncVideoCompositionRequest as AVAsynchronousVideoCompositionRequestMBS)

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Directs a custom video compositor object to create a new pixel buffer composed asynchronously from a collection of sources. (required)
Notes: asyncVideoCompositionRequest: An instance of AVAsynchronousVideoCompositionRequest that provides context for the requested composition.

The custom compositor is expected to invoke, either subsequently or immediately, the asyncVideoCompositionRequest object's finishWithComposedVideoFrame or finishWithError methods.

If you intend to finish rendering the frame after handling of this message returns, you must retain asyncVideoCompositionRequest until after composition is finished.

Note that if the custom compositor's implementation of this method returns without finishing the composition immediately, it may be invoked again with another composition request before the prior request is finished; in such cases the custom compositor should be prepared to manage multiple composition requests.

If the rendered frame is exactly the same as one of the source frames, with no letterboxing, pillboxing or cropping needed, then the appropriate source pixel buffer may be returned, after CFRetain has been called on it).
Available in OS X v10.9 and later.

4.95.7 Properties

4.95.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.95.9 requiredPixelBufferAttributesForRenderContext as Dictionary

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the pixel buffer attributes required by the video compositor for new buffers created for processing. (required) (read-only)
Notes: The property is required to provide a kCVPixelBufferPixelFormatTypeKey key in the dictionary, along with attributes for which the compositor needs specific values to work properly. Omitted attributes will be supplied by the composition engine to allow for the best performance. If the attribute kCVPixelBufferPixelFormatTypeKey key is not in the dictionary an exception will be raised. The value of the kCVPixelBufferPixelFormatTypeKey is an array of kCVPixelFormatType_* constants as defined in Pixel_Format_Types.

The value of requiredPixelBufferAttributesForRenderContext is retrieved prior to the creation of a new render context; the combination of the attributes in the returned value and the additional attributes supplied by the composition engine will be used in the creation of subsequent render context's pixelBuffers.

This property is queried once before any composition request is sent to the compositor. Changing required buffer attributes afterwards is not supported.
Available in OS X v10.9 and later.
(Read only property)

4.95.10 sourcePixelBufferAttributes as Dictionary

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the kinds of source frame pixel buffer attributes a video compositor can accept as input. (required) (read-only)
Notes: The property is required to provide a kCVPixelBufferPixelFormatTypeKey key in the dictionary, along with attributes for which the compositor needs specific values to work properly. Omitted attributes will be supplied by the composition engine to allow for the best performance. If the attribute kCVPixelBufferPixelFormatTypeKey key is not in the dictionary an exception will be raised. The value of the kCVPixelBufferPixelFormatTypeKey is an array of kCVPixelFormatType_* constants as defined in Pixel_Format_Types.

If the custom compositor is meant to be used with an AVVideoCompositionCoreAnimationTool created using the videoCompositionCoreAnimationToolWithAdditionalLayer:asTrackID: method, kCVPixelFormatType_32BGRA should be included as one of the supported pixel format types.

Missing attributes will be set by the composition engine to values allowing the best performance.

This property is queried once before any composition request is sent to the compositor. Changing source buffer attributes afterwards is not supported.
Available in OS X v10.9 and later.
(Read only property)

4.96 class AVVideoCompositionCoreAnimationToolMBS

4.96.1 class AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: You use an AVVideoCompositionCoreAnimationTool object to incorporate Core Animation in a video composition.
Notes: Any animations will be interpreted on the video's timeline, not real-time, so you should:

1. Set animations' beginTime property to 1e-100 rather than 0 (which CoreAnimation replaces with CACurrentMediaTime);
2. Set animations' removedOnCompletion property to false so they are not automatically removed.

4.96.2 Methods

4.96.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.96.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.

4.96.5 videoCompositionCoreAnimationToolWithAdditionalLayer(layer as CALayerMBS, trackID as Integer) as AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Add a Core Animation layer to the video composition.
Notes: layer: The Core Animation layer to add.
trackID: A track ID to identify the track.

trackID should not match any real trackID in the source.

Return a new animation tool for the layer.

You use this method to include a Core Animation layer as an individual track input in video composition. Video composition instructions should reference trackID where the rendered animation should be included.

4.96.6 videoCompositionCoreAnimationToolWithPostProcessingAsVideoLayer(videoLayer as CALayerMBS, animationLayer as CALayerMBS) as AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Composes the composited video frames with a Core Animation layer.
Notes: videoLayer: A video layer.
animationLayer: An animation layer.

Returns a new animation tool for the composition.
Place composited video frames in videoLayer and render animationLayer to produce the final frame.

4.96.7 videoCompositionCoreAnimationToolWithPostProcessingAsVideoLayers(videoLayers() as CALayerMBS, animationLayer as CALayerMBS) as AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Composes the composited video frames with the Core Animation layer.
Notes: videoLayers: An array containing the video layers
animationLayer: The animation layer.

Returns a new AVVideoCompositionCoreAnimationTool instance with the composited video frames and the rendered animation layer.

Duplicates the composited video frames in each videoLayer and renders animationLayer to produce the final frame. The videoLayers should be in animationLayer's sublayer tree.

The animationLayer should not come from, or be added to, another layer tree.
Available in OS X v10.9 and later.

4.96.8 Properties

4.96.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.97 class AVVideoCompositionInstructionMBS

4.97.1 class AVVideoCompositionInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVVideoCompositionInstruction object represents an operation to be performed by a compositor.
Notes: An AVVideoComposition object maintains an array of instructions to perform its composition.

The is a specialty with the AVFoundation framework: There is a class and an interface with same name. The plugin only has one class for both.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.5pr3

4.97.2 Methods

4.97.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.97.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.97.5 copy as AVVideoCompositionInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.97.6 mutableCopy as AVMutableVideoCompositionInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an editable copy of the object.

4.97.7 requiredSourceTrackIDs as Integer()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: An array of video track IDs required to compose frames for this instruction. (required) (read-only)
Notes: Available in OS X v10.9 and later.

4.97.8 Properties

4.97.9 containsTweening as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A Boolean value that returns whether the composition contains tweening. (required) (read-only)
Notes: If YES, rendering a frame from the same source buffers and the same composition instruction at two different compositionTime may yield different output frames. If NO, two such compositions will yield the same frame.

The media pipeline may be able to avoid some duplicate processing when this property is set to false.
Available in OS X v10.9 and later.
(Read only property)

4.97.10 enablePostProcessing as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether post processing is required for the video composition instruction. (read-only)
Notes: A value of false indicates that no post processing is required for the whole duration of the video composition instruction. The composition process is more efficient if the value is false.

The value is true by default.
(Read only property)

4.97.11 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.97.12 passthroughTrackID as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns the track ID if a single source frame should be displayed for the duration of the instruction. (required) (read-only)
Notes: If for the duration of the instruction, the video composition result is one of the source frames, this property returns the corresponding track ID. The compositor won't be run for the duration of the instruction and the proper source frame is used instead.

The dimensions, clean aperture and pixel aspect ratio of the source buffer are matched to the required values automatically.
Available in OS X v10.9 and later.
(Read only property)

4.97.13 timeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range during which the instruction is effective. (read-only)
Notes: If the time range is invalid, the video compositor will ignore it.
(Read only property)

4.98 class AVVideoCompositionLayerInstructionMBS

4.98.1 class AVVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVVideoCompositionLayerInstruction object represents the transform and opacity ramps to apply to a given track.

4.98.2 Methods

4.98.3 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.98.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The default constructor.

4.98.5 copy as AVVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.98.6 getCropRectangleRampForTime(time as CMTimeMBS, byref startCropRectangle as CGRectMBS, byref endCropRectangle as CGRectMBS, byref timeRange as CMTimeRangeMBS) as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Obtains the crop rectangle ramp that includes the specified time.
Notes: time: If a ramp with a time range that contains the specified time has been set, information about the effective ramp for that time is supplied. Otherwise, information about the first ramp that starts after the specified time is supplied.

startCropRectangle: A CGRect to receive the starting crop rectangle value for the crop rectangle ramp.
endCropRectangle: A CGRect to receive the ending crop rectangle value for the crop rectangle ramp.
timeRange: A CMTimeRange to receive the time range of the crop rectangle ramp.

Returns false will be returned if the specified time is beyond the duration of the last crop rectangle ramp that has been set.
Available in OS X v10.9 and later.

4.98.7 getOpacityRampForTime(time as CMTimeMBS, byref startOpacity as Double, byref endOpacity as Double, byref timeRange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Obtains the opacity ramp that includes a specified time.
Notes: time: If a ramp with a time range that contains the specified time has been set, information about the effective ramp for that time is supplied. Otherwise, information about the first ramp that starts after the specified time is supplied.
startOpacity: a float to receive the starting opacity value for the opacity ramp.
endOpacity: a float to receive the ending opacity value for the opacity ramp.
timeRange: a CMTimeRange to receive the time range of the opacity ramp.

Returns true if values are returned successfully, otherwise false. False is returned if time is beyond the duration of the last opacity ramp that has been set.

4.98.8 getTransformRampForTime(time as CMTimeMBS, byref startTransform as CGAffineTransformMBS, byref endOpacity as CGAffineTransformMBS, byref timeRange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Obtains the transform ramp that includes a specified time.
Notes: time
If a ramp with a time range that contains the specified time has been set, information about the effective ramp for that time is supplied. Otherwise, information about the first ramp that starts after the specified time is supplied.
startTransform: A float to receive the starting transform value for the transform ramp.
endTransform: A float to receive the ending transform value for the transform ramp.
timeRange: A CMTimeRange to receive the time range of the transform ramp.
Returns true if values are returned successfully, otherwise false. False is returned if time is beyond the duration of the last transform ramp that has been set.

4.98.9 mutableCopy as AVMutableVideoCompositionLayerInstructionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an editable copy of the object.

4.98.10 trackID as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The trackID of the source track to which the compositor will apply the instruction. (read-only)

4.98.11 Properties

4.98.12 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.99 class AVVideoCompositionMBS

4.99.1 class AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: An AVVideoComposition object represents an immutable video composition.
Notes: The AVFoundation framework also provides a mutable subclass, AVMutableVideoComposition, that you can use to create new videos.

4.99.2 Methods

4.99.3 animationTool as AVVideoCompositionCoreAnimationToolMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A video composition tool to use with Core Animation in offline rendering. (read-only)
Notes: This attribute may be nil.

You set an animation tool if you are using the composition in conjunction with AVAssetExportSession for offline rendering, rather than with AVPlayer.

4.99.4 available as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer.

4.99.5 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The constructor.

4.99.6 copy as AVVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.99.7 frameDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The interval for which the video composition should render composed video frames. (read-only)
Notes: This property only applies when the composition is enabled.

4.99.8 instructions as AVVideoCompositionInstructionMBS()

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The video composition instructions. (read-only)
Notes: The array contains of instances of AVVideoCompositionInstruction.

For the first instruction in the array, timeRange.start must be less than or equal to the earliest time for which playback or other processing will be attempted (typically kCMTimeZero). For subsequent instructions, timeRange.start must be equal to the prior instruction's end time. The end time of the last instruction must be greater than or equal to the latest time for which playback or other processing will be attempted (typically be the duration of the asset with which the instance of AVVideoComposition is associated).

4.99.9 isValidForAsset(asset as AVAssetMBS, timerange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether the time ranges of the composition's instructions conform to validation requirements.
Notes: asset: An AVAsset object, if you wish to validate the time ranges of the instructions against the duration of the asset and the track IDs of the layer instructions against the asset's tracks. Pass nil to skip that validation.
timeRange: A time range.
Only those instructions with time ranges that overlap with this time range will be validated. To validate all instructions that may be used for playback or other processing, regardless of time range, pass CMTimeRangeMake(kCMTimeZero, kCMTimePositiveInfinity).

Returns true if the time ranges of the composition's instructions conform to validation requirements, otherwise false.

This method may call events videoCompositionShouldContinueValidatingAfterFindingInvalidValueForKey, videoCompositionShouldContinueValidatingAfterFindingInvalidTrackIDInInstruction, videoCompositionShouldContinueValidatingAfterFindingInvalidTimeRangeInInstruction or videoCompositionShouldContinueValidatingAfterFindingEmptyTimeRange on AVFoundationMBS class.

In the course of validation, the receiver will invoke its events (if there is one) with reference to any trouble spots in the video composition.

This method raises an exception if the event modifies the receiver's array of instructions or the array of layer instructions of any AVVideoCompositionInstruction object contained therein during validation.

4.99.10 mutableCopy as AVMutableVideoCompositionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates an editable copy of the object.

4.99.11 renderSize as CGSizeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The size at which the video composition should render. (read-only)
Notes: This property only applies when the composition is enabled.

4.99.12 videoCompositionWithPropertiesOfAsset(asset as AVAssetMBS) as AVVideoCompositionMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Creates and returns a video composition object configured to present the video tracks of the specified asset.
Notes: asset: The asset whose configuration matches the intended use of the video composition.

Returns a new video composition object.

This method creates the video composition object and configures it with the values and instructions suitable for presenting the video tracks of the specified asset. The returned object contains instructions that respect the spatial properties and time ranges of the specified asset's video tracks. It also configures the object properties in the following way:

- The value of the frameDuration property is set to a value short enough to accommodate the greatest nominal frame rate value among the asset's video tracks, as indicated by the nominalFrameRate property of each track. If all of the asset tracks have a nominal frame rate of 0, a frame rate of 30 frames per second is used, with the frame duration set accordingly.
- The value assigned to the renderSize property depends on whether the asset is an AVComposition object. If it is, the value is set to the value of the naturalSize property of the composition. If it is not, the value is set to a value that encompasses all of the asset's video tracks.
- The value of the renderScale property is set to 1.0.
- The value of the animationTool property is set to nil.

Available in OS X v10.9 and later.

4.99.13 Properties

4.99.14 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.100 class AVVideoCompositionRenderContextMBS

4.100.1 class AVVideoCompositionRenderContextMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The AVVideoCompositionRenderContext class defines the context within which custom compositors render new output pixels buffers.
Notes: An instance of AVVideoCompositionRenderContext provides size and scaling information and offers a service for efficiently providing pixel buffers from a managed pool of buffers.
Available in OS X v10.9 and later.

4.100.2 Methods

4.100.3 available as boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.9 or later.

4.100.4 Constructor

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The constructor.

4.100.5 newPixelBuffer as CVPixelBufferMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Returns a pixel buffer to use for rendering.
Notes: The buffer's kCVImageBufferCleanApertureKey and kCVImageBufferPixelAspectRatioKey attachments are set to match the current composition processor properties.
You are responsible for calling CVBufferRelease on the pixel buffer.
Available in OS X v10.9 and later.

4.100.6 Properties

4.100.7 edgeWidths as AVEdgeWidthsMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The thickness of the edge processing region on the left, top, right and bottom edges, in pixels. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.100.8 Handle as Integer

Plugin Version: 14.0, Platform: macOS, Targets: All.

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

4.100.9 highQualityRendering as Boolean

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The rendering quality to use. (read-only)
Notes: Specifies that the custom compositor should use higher quality, potentially slower algorithms.
Generally this property is true for non-real-time use cases.
Available in OS X v10.9 and later.
(Read only property)

4.100.10 pixelAspectRatio as AVPixelAspectRatioMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The pixel aspect ratio for rendered frames. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.100.11 renderScale as Double

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: A scaling ratio that is applied when rendering frames. (read-only)

Notes: Available in OS X v10.9 and later.
(Read only property)

4.100.12 renderTransform as CGAffineTransformMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: Transform to apply to the source image (read-only)
Notes: The transform to apply to the source image incorporating the renderScale, pixelAspectRatio, and edgeWidths.
The coordinate system origin is the top left corner of the buffer.
Available in OS X v10.9 and later.
(Read only property)

4.100.13 size as CGSizeMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The width and height for rendering frames. (read-only)
Notes: Available in OS X v10.9 and later.
(Read only property)

4.100.14 videoComposition as AVVideoCompositionMBS

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The video composition being rendered.
Notes: Available in OS X v10.9 and later.
(Read only property)

4.101 class CMFormatDescriptionMBS

4.101.1 class CMFormatDescriptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The class for a format description.
Notes: CMFormatDescriptions are immutable Core Foundation objects that describe media data of various types, including audio, video, and muxed media data. There are two types of API: media-type-agnostic APIs (supported by all CMFormatDescriptions) and media-type-specific APIs. The media-type-agnostic APIs are prefixed with CMFormatDescription, and the media-type-specific APIs are prefixed with CMAudioFormat-Description CMVideoFormatDescription, and so on.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.101.2 Methods

4.101.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.101.4 Equal(other as CMFormatDescriptionMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Checks if two descriptions are equal.

4.101.5 Extensions as Dictionary

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Queries extensions to the description.

4.101.6 MediaSubType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Queries media sub type.

4.101.7 MediaType as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Queries media type.

4.101.8 Name as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Queries the name of this format description.

4.101.9 Properties

4.101.10 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.102 class CMSampleBufferMBS

4.102.1 class CMSampleBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: CMSampleBuffers are CF objects containing zero or more compressed (or uncompressed) samples of a particular media type (audio, video, muxed, etc), that are used to move media sample data through the media system.
Notes: A CMSampleBuffer can contain a CMBlockBuffer of one or more media samples or a CVImageBuffer, a reference to the format description for the stream of CMSampleBuffers, size and timing information for each of the contained media samples, and both buffer-level and sample-level attachments. The buffer-level attachments of a CMSampleBuffer are distinct from the attachments of its contained CMBlockBuffer. An example of a sample-level attachment is an annotation about video frame dependencies (eg. "droppable", "other frames depend on me", "I depend on other frames", etc). Each sample (video frame) in the CMSampleBuffer would need its own attachment in this case. Another sample-level attachment example is SMPTE timecode acquired during capture. To get and set a CMSampleBuffer's buffer-level attachments, use the APIs in in CMAttachmentBearer.h (CMGetAttachment et al).
It is possible for a CMSampleBuffer to describe samples it does not yet contain. For example, some media services may have access to sample size, timing and format information before the data is read. Such services may create CMSampleBuffers with that information and insert them into queues early, and attach (or fill) the CMBlockBuffers of media data later, when the data becomes ready. To this end, CMSampleBuffers have the concept of data-readiness, which can be tested, set, forced to become ready "now", etc. It is also possible for a CMSampleBuffer to contain nothing but a special buffer-level attachment that describes a media stream event (eg. "discontinuity: drain and reset decoder before processing the next CMSampleBuffer"). Such a special attachment can also be attached to regular CMSampleBuffers (ie. that contain media sample data), and if so, the event it describes is defined to occur after the samples in that CMSampleBuffer.

The MBS Plugin implements a subset of what's available. If you need more, please do not hesitate to contact us.
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 18.4pr3
- AVCaptureVideoDataOutput improvements
- MBS Xojo / Real Studio Plugins, version 15.4pr2
- MBS Xojo / Real Studio Plugins, version 14.5pr2

4.102.2 Methods

4.102.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.102.4 Copy as CMSampleBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

4.102.5 CopySampleBufferForRange(pos as Integer, len as Integer) as CMSampleBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a CMSampleBuffer containing a range of samples from an existing CMSampleBuffer.
Notes: Samples containing non-interleaved audio are currently not supported.

4.102.6 Invalidate

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Makes the sample buffer invalid, calling any installed invalidation callback.
Notes: An invalid sample buffer cannot be used – all accessors will return kCMSampleBufferError_Invalidated.
It is not a good idea to do this to a sample buffer that another module may be accessing concurrently.
Example of use: the invalidation callback could cancel pending I/O.

4.102.7 MakeDataReady

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Makes a CMSampleBuffer's data ready, by calling the client's MakeDataReadyCallback.
Notes: The CMSampleBufferMakeDataReadyCallback is passed in by the client during creation. It must return 0 if successful, and in that case, CMSampleBufferMakeDataReady will set the data readiness of the CMSampleBuffer to true. Example of usage: when it is time to actually use the data. Example of callback routine: a routine to force a scheduled read to complete. If the CMSampleBuffer is not ready, and there is no CMSampleBufferMakeDataReadyCallback to call, kCMSampleBufferError_BufferNotReady will be returned. Similarly, if the CMSampleBuffer is not ready, and the CMSampleBufferMakeDataReadyCallback fails and returns an error, kCMSampleBufferError_BufferNotReady will be returned.

4.102.8 SampleSize(index as Integer) as UInt64

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Returns the size in bytes of a specified sample in a CMSampleBuffer.
Notes: Size in bytes of the specified sample in the CMSampleBuffer.
If the sample index is not in the range 0 .. numSamples-1, a size of 0 will be returned. If there are no sample sizes in this CMSampleBuffer, a size of 0 will be returned. This will be true, for example, if the samples in the buffer are non-contiguous (eg. non-interleaved audio, where the channel values for a single sample are scattered through the buffer), or if this CMSampleBuffer contains a CVImageBuffer.

4.102.9 SetDataReady

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Marks a CMSampleBuffer's data as "ready".
Notes: There is no way to undo this operation. The only way to get an "unready" CMSampleBuffer is to call CMSampleBufferCreate with the dataReady parameter set to false. Example of usage: in a read completion routine.

4.102.10 Properties

4.102.11 DataIsReady as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns whether or not a CMSampleBuffer's data is ready.
Notes: Whether or not the CMSampleBuffer's data is ready. True is returned for special marker buffers, even though they have no data. False is returned if there is an error.
(Read only property)

4.102.12 DecodeTimeStamp as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the numerically earliest decode timestamp of all the samples in a CMSampleBuffer.
Notes: The returned decode timestamp is always the decode timestamp of the first sample in the buffer, since even out-of-presentation-order samples are expected to be in decode order in the buffer.
Numerically earliest sample decode timestamp in the CMSampleBuffer. kCMTimeInvalid is returned if there is an error.
(Read only property)

4.102.13 Duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the total duration of a CMSampleBuffer.
Notes: If the buffer contains out-of-presentation-order samples, any gaps in the presentation timeline are not represented in the returned duration.
The returned duration is simply the sum of all the individual sample durations.
Returns the duration of the CMSampleBuffer. kCMTimeInvalid is returned if there is an error.
(Read only property)

4.102.14 FormatDescription as CMFormatDescriptionMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the format description of the samples in a CMSampleBuffer.
Notes: Nil is returned if there is an error.
(Read only property)

4.102.15 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.102.16 ImageBuffer as CVImageBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a CMSampleBuffer's CVImageBuffer of media data.
Notes: The result will be nil if the CMSampleBuffer does not contain a CVImageBuffer, if the CMSampleBuffer contains a CMBlockBuffer, or if there is some other error.
(Read only property)

4.102.17 IsValid as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Queries whether a sample buffer is still valid.
Notes: Returns false if buffer is nil or Invalidate was called, true otherwise.
Does not perform any kind of exhaustive validation of the sample buffer.

(Read only property)

4.102.18 Lasterror as Integer

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The last error code.
Notes: See error code constants.
Looking with debugger on properties of this class will change the lasterror code!
(Read and Write property)

4.102.19 NumberOfSamples as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the number of media samples in a CMSampleBuffer.
Notes: The number of media samples in the CMSampleBuffer. 0 is returned if there is an error.
(Read only property)

4.102.20 OutputDecodeTimeStamp as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the output decode timestamp of the CMSampleBuffer.
Notes: For consistency with OutputPresentationTimeStamp, this is calculated as: OutputPresentationTimeStamp + ((DecodeTimeStamp - PresentationTimeStamp) / SpeedMultiplier).
CMInvalidTime is returned if there is an error.
(Read only property)

4.102.21 OutputDuration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the output duration of a CMSampleBuffer.
Notes: The OutputDuration is the duration minus any trimmed duration, all divided by the SpeedMultiplier: (Duration - TrimDurationAtStart - TrimDurationAtEnd) / SpeedMultiplier
Returns the output duration of the CMSampleBuffer. kCMTimeInvalid is returned if there is an error.
(Read only property)

4.102.22 OutputPresentationTimeStamp as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the output presentation timestamp of the CMSampleBuffer.
Notes: The output presentation timestamp is the time at which the decoded, trimmed, stretched and possibly reversed samples should commence being presented.
If CMSampleBufferSetOutputPresentationTimeStamp has been called to explicitly set the output PTS, CMSampleBufferGetOutputPresentationTimeStamp returns it.
If not, CMSampleBufferGetOutputPresentationTimeStamp calculates its result as (PresentationTimeStamp + TrimDurationAtStart) unless kCMSampleBufferAttachmentKey_Reverse is kCFBooleanTrue, in which case it calculates the result as (PresentationTimeStamp + Duration - TrimDurationAtEnd).
These are generally correct for un-stretched, un-shifted playback.
For general forward playback in a scaled edit, the OutputPresentationTimeStamp should be set to:
((PresentationTimeStamp + TrimDurationAtStart - EditStartMediaTime) / EditSpeedMultiplier) + EditStartTrackTime.
For general reversed playback:
((PresentationTimeStamp + Duration - TrimDurationAtEnd - EditStartMediaTime) / EditSpeedMultiplier) + EditStartTrackTime.
Returns kCMTimeInvalid is returned if there is an error.
(Read and Write property)

4.102.23 PresentationTimeStamp as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the numerically earliest presentation timestamp of all the samples in a CMSampleBuffer.
Notes: For in-presentation-order samples, this is the presentation timestamp of the first sample.
For out-of-presentation-order samples, this is the presentation timestamp of the sample that will be presented first, which is not necessarily the first sample in the buffer.
Returns numerically earliest sample presentation timestamp in the CMSampleBuffer. kCMTimeInvalid is returned if there is an error.
(Read only property)

4.102.24 Text as String

Plugin Version: 15.4, Platform: macOS, Targets: All.

Function: For text samples, provides the text.
Notes: (Read only property)

4.102.25 TotalSampleSize as UInt64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the total size in bytes of sample data in a CMSampleBuffer.
Notes: If there are no sample sizes in this CMSampleBuffer, a size of 0 will be returned.
(Read only property)

4.102.26 Constants

Error Codes

Constant	Value	Description
kAllocationFailed	-12730	An allocation failed.
kAlreadyHasDataBuffer	-12732	Attempt was made to set a dataBuffer on a CMSampleBuffer that already ha one.
kArrayTooSmall	-12737	Output array was not large enough for the array being requested.
kBufferHasNoSampleSizes	-12735	Attempt to get sample size information when there was none.
kBufferHasNoSampleTimingInfo	-12736	Attempt to get sample timing information when there was none.
kBufferNotReady	-12733	Buffer could not be made ready.
kCannotSubdivide	-12739	Sample buffer does not contain sample sizes. This can happen when the sample in the buffer are non-contiguous (eg. non-interleaved audio, where the channe values for a single sample are scattered through the buffer).
kInvalidated	-12744	the sample buffer was invalidated.
kInvalidEntryCount	-12738	Timing info or size array entry count was not 0, 1, or numSamples.
kInvalidMediaFormat	-12743	The format of the given media does not match the given format descriptio (eg. a format description paired with a CVImageBuffer that fails CMVideo FormatDescriptionMatchesImageBuffer).
kInvalidMediaTypeForOperation	-12741	The media type specified by a format description is not valid for the give operation (eg. a CMSampleBuffer with a non-audio format description passe to GetAudioStreamPacketDescriptions).
kInvalidSampleData	-12742	Buffer contains bad data. Only returned by CMSampleBuffer functions tha inspect its sample data.
kRequiredParameterMissing	-12731	Nil or 0 was passed for a required parameter.
kSampleIndexOutOfRange	-12734	Sample index was not between 0 and numSamples-1, inclusive.
kSampleTimingInfoInvalid	-12740	Buffer unexpectedly contains a non-numeric sample timing info.

4.103 class CMTimeMappingMBS

4.103.1 class CMTimeMappingMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A class to specify the mapping of a segment of one time line into another.
Notes: A CMTimeMapping specifies the mapping of a segment of one time line (called the source) into another time line (called the target). When used for movie edit lists, the source time line is the media and the target time line is the track or movie.

4.103.2 Methods

4.103.3 Constructor(source as CMTimeRangeMBS, target as CMTimeRangeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Initializes the time mapping object.

4.103.4 Properties

4.103.5 Source as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range on the source time line.
Notes: For an empty edit, source.start is an invalid CMTime, in which case source.duration is ignored. Otherwise, source.start is the starting time within the source, and source.duration is the duration of the source timeline to be mapped to the target time range.
(Read and Write property)

4.103.6 Target as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The time range on the target time line.
Notes: If target.duration and source.duration are different, then the source segment should be played at the rate source.duration /target.duration to fit.
(Read and Write property)

4.104 class CMTimeMBS

4.104.1 class CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: CMTime is a class representing times (either timestamps or durations).
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox t.Description
```

Notes: A CMTime is represented as a rational number, with a numerator (an int64 value), and a denominator (an int32 timescale). Conceptually, the timescale specifies the fraction of a second each unit in the numerator occupies. Thus if the timescale is 4, each unit represents a quarter of a second; if the timescale is 10, each unit represents a tenth of a second, and so on. In addition to a simple time value, a CMTime can represent non-numeric values: +infinity, -infinity, and indefinite. Using a flag CMTime indicates whether the time been rounded at some point.

CMTimes contain an epoch number, which is usually set to 0, but can be used to distinguish unrelated timelines: for example, it could be incremented each time through a presentation loop, to differentiate between time N in loop 0 from time N in loop 1.

Additional functions for managing dates and times are described in Time Utilities Reference. Note that CMTime is designed for media timelines whereas functions in Time Utilities Reference are designed for working with wall-clock time in Core Foundation framework; see also AV Foundation Constants Reference.
Blog Entries

- Automated video editing with Xojo

Xojo Developer Magazine

- 12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.104.2 Methods

4.104.3 AbsoluteValue as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the absolute value of a CMTime.
Notes: Same as the argument time, with sign inverted if negative.

4.104.4 Add(other as CMTimeMBS) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the sum of two CMTimes.
Notes: self: CMTime to be added.
other: Another CMTime to be added.
Returns the sum of the two CMTimes (addend1 + addend2).

If the operands both have the same timescale, the timescale of the result will be the same as the operands' timescale. If the operands have different timescales, the timescale of the result will be the least common multiple of the operands' timescales. If that LCM timescale is greater than kCMTimeMaxTimescale, the result timescale will be kCMTimeMaxTimescale, and default rounding will be applied when converting the result to this timescale. If the result value overflows, the result timescale will be repeatedly halved until the result value no longer overflows. Again, default rounding will be applied when converting the result to this timescale. If the result value still overflows when timescale == 1, then the result will be either positive or negative infinity, depending on the direction of the overflow.

If any rounding occurs for any reason, the result's kCMTimeFlags_HasBeenRounded flag will be set. This flag will also be set if either of the operands has kCMTimeFlags_HasBeenRounded set. If either of the operands is invalid, the result will be invalid. If the operands are valid, but just one operand is infinite, the result will be similarly infinite. If the operands are valid, and both are infinite, the results will be as follows:

+infinity + +infinity == +infinity
-infinity + -infinity == -infinity

+infinity + -infinity == invalid

-infinity + +infinity == invalid

If the operands are valid, not infinite, and either or both is indefinite, the result will be indefinite. If the two operands are numeric (i.e.. valid, not infinite, not indefinite), but have different nonzero epochs, the result will be invalid. If they have the same nonzero epoch, the result will have epoch zero (a duration). Times in different epochs cannot be added or subtracted, because epoch length is unknown. Times in epoch zero are considered to be durations and can be added to times in other epochs. Times in different epochs can be compared, however, because numerically greater epochs always occur after numerically lesser epochs.

4.104.5 Compare(other as CMTimeMBS) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the numerical relationship of two CMTimes.
Notes: self: First CMTime in comparison.

other: Second CMTime in comparison.

Returns the numerical relationship of the two CMTimes.

-1 is returned if time1 is less than time2.
1 is returned if time1 is greater than time2.
0 is returned if time1 and time2 are equal.

If the two CMTimes are numeric (i.e.. not invalid, infinite, or indefinite), and have different epochs, it is considered that times in numerically larger epochs are always greater than times in numerically smaller epochs. Since this routine will be used to sort lists by time, it needs to give all values (even invalid and indefinite ones) a strict ordering to guarantee that sort algorithms terminate safely. The order chosen is somewhat arbitrary: -infinity <all finite values <indefinite <+infinity <invalid

Invalid CMTimes are considered to be equal to other invalid CMTimes, and greater than any other CMTime. Positive infinity is considered to be less than any invalid CMTime, equal to itself, and greater than any other CMTime. An indefinite CMTime is considered to be less than any invalid CMTime, less than positive infinity, equal to itself, and greater than any other CMTime. Negative infinity is considered to be equal to itself, and less than any other CMTime.

4.104.6 Constructor(Value as Int64, Timescale as Integer, Flags as Integer = 1, Epoch as Int64 = 0)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a new time object with the given field values.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox t.Description
```

4.104.7 ConvertScale(newTimescale as Integer, RoundingMethod as Integer = 1) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a new CMTime containing the source CMTime converted to a new timescale (rounding as requested).
Example:

```
dim t as CMTimeMBS = CMTimeMBS.MakeWithSeconds(123.4, 600)
dim u as CMTimeMBS = t.ConvertScale(300, t.kCMTimeRoundingMethod_Default)
```

```
MsgBox t.Description+EndOfLine+u.Description
```

Notes: self: Source CMTime.
newTimescale: The timescale to which the source CMTime is converted.
RoundingMethod: The requested rounding method.

Returns the converted CMTime.
If the value needs to be rounded, the kCMTimeFlags_HasBeenRounded flag will be set. See constants for a discussion of the various rounding methods available. If the source time is non-numeric (i.e.. infinite, indefinite, invalid), the result will be similarly non-numeric.

4.104.8 Description as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a description of the time.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox t.Description
```

4.104.9 kCMTimeIndefinite as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Predefined time.
Notes: Use this constant to initialize an indefinite CMTime (for example, the duration of a live broadcast). Do not test against this using (time = kCMTimeIndefinite), there are many CMTimes other than this that are also indefinite. Use time.IsIndefinite instead.

4.104.10 kCMTimeInvalid as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Predefined time.
Example:

```
dim t as CMTimeMBS = CMTimeMBS.kCMTimeInvalid
MsgBox "Invalid: "+str(t.IsInvalid)
```

Notes: Use this constant to initialize an invalid CMTime.
All fields are 0. Do not test against this using (time = kCMTimeInvalid), there are many CMTimes other than this that are also invalid. Use time.IsInvalid instead.

4.104.11 kCMTimeNegativeInfinity as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Predefined time.
Notes: Use this constant to initialize a CMTime to -infinity.
Do not test against this using (time = kCMTimeNegativeInfinity), there are many CMTimes other than this that are also -infinity. Use time.IsNegativeInfinity instead.

4.104.12 kCMTimePositiveInfinity as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Predefined time.
Notes: Use this constant to initialize a CMTime to +infinity.
Do not test against this using (time = kCMTimePositiveInfinity), there are many CMTimes other than this that are also +infinity. Use time.IsPositiveInfinity instead.

4.104.13 kCMTimeZero as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Predefined time.
Notes: Use this constant to initialize a CMTime to 0.
Do not test against this using (time = kCMTimeZero), there are many CMTimes other than this that are also 0. Use time.Compare(kCMTimeZero) instead.

4.104.14 Make(value as Int64, timescale as Integer) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Makes a valid CMTime with value and timescale.
Example:

```
dim t as CMTimeMBS = CMTimeMBS.Make(1234, 600)
MsgBox t.Description
```

Notes: Epoch is implied to be 0.

value: Initializes the value field of the resulting CMTime.
timescale: Initializes the timescale field of the resulting CMTime.

Returns the resulting CMTime

4.104.15 MakeWithEpoch(value as Int64, timescale as Integer, Epoch as Int64) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Makes a valid CMTime with value, scale and epoch.
Notes: value: Initializes the value field of the resulting CMTime.
timescale: Initializes the scale field of the resulting CMTime.
epoch: Initializes the epoch field of the resulting CMTime.

Returns the resulting CMTime.

4.104.16 MakeWithSeconds(seconds as Double, preferredTimeScale as Int32 = 600) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Makes a CMTime from a Float number of seconds, and a preferred timescale.
Example:

```
dim t as CMTimeMBS = CMTimeMBS.MakeWithSeconds(123.4, 600)
MsgBox t.Description
```

Notes: seconds: Initializes the seconds field of the resulting CMTime.
preferredTimeScale: Initializes the preferredTimeScale field of the resulting CMTime.

Returns the resulting CMTime.

The epoch of the result will be zero. If preferredTimeScale is <= 0, the result will be an invalid CMTime. If the preferred timescale will cause an overflow, the timescale will be halved repeatedly until the overflow goes away, or the timescale is 1. If it still overflows at that point, the result will be +/- infinity. The kCMTimeFlags_HasBeenRounded flag will be set if the result, when converted back to seconds, is not exactly equal to the original seconds value.

4.104.17 Maximum(t1 as CMTimeMBS, t2 as CMTimeMBS) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the greater of two CMTimes.
Example:

```
dim t1 as new CMTimeMBS(1200, 600)
dim t2 as new CMTimeMBS(70, 30)

dim mi as CMTimeMBS = CMTimeMBS.Minimum(t1,t2)
dim ma as CMTimeMBS = CMTimeMBS.Maximum(t1,t2)

MsgBox "with "+t1.Description+_
" and "+t2.description+_
" the minimum is "+mi.Description+_
" and maximum is "+ma.Description
```

4.104.18 Minimum(t1 as CMTimeMBS, t2 as CMTimeMBS) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the lesser of two CMTimes.
Example:

```
dim t1 as new CMTimeMBS(1200, 600)
dim t2 as new CMTimeMBS(70, 30)

dim mi as CMTimeMBS = CMTimeMBS.Minimum(t1,t2)
dim ma as CMTimeMBS = CMTimeMBS.Maximum(t1,t2)

MsgBox "with "+t1.Description+_
" and "+t2.description+_
" the minimum is "+mi.Description+_
" and maximum is "+ma.Description
```

4.104.19 Multiply(multiplier as Integer) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the product of a CMTime and a 32-bit integer.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox str(T.Seconds)+" seconds"
t = t.Multiply(2)
MsgBox str(T.Seconds)+" seconds"
```

Notes: self: The CMTime that will be multiplied.
multiplier: A 32-bit integer. CMTime time and multiplier will be multiplied.

Returns the product of the CMTime and the 32-bit integer.

The result will have the same timescale as the CMTime operand. If the result value overflows, the result timescale will be repeatedly halved until the result value no longer overflows. Again, default rounding will be applied when converting the result to this timescale. If the result value still overflows when timescale == 1, then the result will be either positive or negative infinity, depending on the direction of the overflow. If any rounding occurs for any reason, the result's kCMTimeFlags_HasBeenRounded flag will be set. This flag will also be set if the CMTime operand has kCMTimeFlags_HasBeenRounded set. If the CMTime operand is invalid, the result will be invalid. If the CMTime operand is valid, but infinite, the result will be infinite, and of an appropriate sign, give the signs of both operands. If the CMTime operand is valid, but indefinite, the result will be indefinite.

4.104.20 MultiplyByFloat(multiplier as Double) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the product of a CMTime and a 64-bit float.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox str(T.Seconds)+" seconds"
t = t.MultiplyByFloat(2.0)
MsgBox str(T.Seconds)+" seconds"
```

Notes: time: The CMTime that will be multiplied.
multiplier: A 64-bit float. CMTime time and multiplier will be multiplied.

Returns the product of the CMTime and the 64-bit float.

The result will initially have the same timescale as the CMTime operand. If the result timescale is less than 65536, it will be repeatedly doubled until it is at least 65536. If the result value overflows, the result timescale will be repeatedly halved until the result value no longer overflows. Again, default rounding will be applied when converting the result to this timescale. If the result value still overflows when timescale == 1, then the result will be either positive or negative infinity, depending on the direction of the overflow. If any rounding occurs for any reason, the result's kCMTimeFlags_HasBeenRounded flag will be set. This flag will also be set if the CMTime operand has kCMTimeFlags_HasBeenRounded set. If the CMTime operand is invalid, the result will be invalid. If the CMTime operand is valid, but infinite, the result will be infinite, and of an appropriate sign, given the signs of both operands. If the CMTime operand is valid, but indefinite, the result will be indefinite.

4.104.21 Show

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prints a description of the CMTime.
Example:

```
dim t as new CMTimeMBS(1200, 600)
t.show
```

4.104.22 Subtract(other as CMTimeMBS) as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the difference of two CMTimes.
Notes: self: The CMTime from which the subtrahend will be subtracted.
other: The CMTime that will be subtracted from the minuend.

Returns the difference of the two CMTimes (minuend - subtrahend).

If the operands both have the same timescale, the timescale of the result will be the same as the operands' timescale. If the operands have different timescales, the timescale of the result will be the least common multiple of the operands' timescales. If that LCM timescale is greater than kCMTimeMaxTimescale, the result timescale will be kCMTimeMaxTimescale, and default rounding will be applied when converting the result to this timescale.

If the result value overflows, the result timescale will be repeatedly halved until the result value no longer overflows. Again, default rounding will be applied when converting the result to this timescale. If the result value still overflows when timescale = 1, then the result will be either positive or negative infinity, depending on the direction of the overflow.

If any rounding occurs for any reason, the result's kCMTimeFlags_HasBeenRounded flag will be set. This flag will also be set if either of the operands has kCMTimeFlags_HasBeenRounded set. If either of the operands is invalid, the result will be invalid. If the operands are valid, but just one operand is infinite, the result will be similarly infinite. If the operands are valid, and both are infinite, the results will be as follows:

+infinity - +infinity == invalid
-infinity - -infinity == invalid
+infinity - -infinity == +infinity
-infinity - +infinity == -infinity

If the operands are valid, not infinite, and either or both is indefinite, the result will be indefinite. If the two operands are numeric (i.e.. valid, not infinite, not indefinite), but have different nonzero epochs, the result will be invalid. If they have the same nonzero epoch, the result will have epoch zero (a duration). Times in different epochs cannot be added or subtracted, because epoch length is unknown. Times in epoch zero are considered to be durations and can be subtracted from times in other epochs. Times in different epochs can be compared, however, because numerically greater epochs always occur after numerically lesser epochs.

4.104.23 Properties

4.104.24 Epoch as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The epoch of the CMTime.
Notes: You use the epoch to differentiate between equal timestamps that are actually different because of looping, multi-item sequencing, and so on.
The epoch is used during comparison: greater epochs happen after lesser ones. Addition or subtraction is only possible within a single epoch, however, since the epoch length may be unknown or variable.

The epoch is typically 0, but you might use a different value, for example, in a loop.
(Read and Write property)

4.104.25 Flags as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A bitfield representing the flags set for the CMTime.
Notes: For example, kCMTimeFlags_Valid. See constants for possible values.
(Read and Write property)

4.104.26 HasBeenRounded as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time has been rounded.
Notes: Returns true if the CMTime has been rounded, otherwise false (the time is completely accurate).
(Read only property)

4.104.27 IsIndefinite as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns whether a CMTime is indefinite.
Notes: Returns true if the CMTime is indefinite, false if it is not.
(Read only property)

4.104.28 IsInvalid as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time is invalid.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox "Invalid: "+str(t.IsInvalid)

t = CMTimeMBS.kCMTimeInvalid
MsgBox "Invalid: "+str(t.IsInvalid)
```

Notes: Returns true if the CMTime is invalid, otherwise false.
(Read only property)

4.104.29 IsNegativeInfinity as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time is negative infinity.
Notes: Returns true if the CMTime is negative infinity, otherwise false.

Use this instead of (myTime = kCMTimeNegativeInfinity), since there are many CMTime structs that represent positive infinity. This is because the non-flags fields are ignored, so they can contain anything.

(Read only property)

4.104.30 IsNumeric as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time is numeric.
Notes: Returns true if the CMTime is numeric, otherwise false. Returns false if the CMTime is invalid, indefinite, or +/- infinity.
A numeric time contains a usable value/timescale/epoch.
(Read only property)

4.104.31 IsPositiveInfinity as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time is positive infinity.
Notes: Returns true if the CMTime is positive infinity, otherwise false.

Use this instead of (myTime = kCMTimePositiveInfinity), since there are many CMTime structs that represent positive infinity. This is because the non-flags fields are ignored, so they can contain anything.
(Read only property)

4.104.32 IsValid as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given time is valid.
Example:

```
dim t as new CMTimeMBS(1200, 600)
MsgBox "valid: "+str(t.IsValid)
```

Notes: Return true if the CMTime is valid, otherwise false.
(Read only property)

4.104.33 Seconds as Double

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The number of seconds.
Example:

```
dim t as new CMTimeMBS(1200, 600)
t.show
```

Notes: If the CMTime is invalid or indefinite, NaN is returned. If the CMTime is infinite, +/- infinity is returned. If the CMTime is numeric, epoch is ignored, and time.value / time.timescale is returned. The division is done in Float64, so the fraction is not lost in the returned result.
(Read only property)

4.104.34 Timescale as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The timescale of the CMTime.
Notes: value/timescale = seconds.
(Read and Write property)

4.104.35 Value as Int64

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The value of the CMTime.
Notes: value/timescale = seconds.
(Read and Write property)

4.104.36 Constants

Constants

Constant	Value	Description
kCMTimeMaxTimescale	&h7fffffff	A constant to define the maximum timescale.

Flags

Constant	Value	Description
kCMTimeFlags_HasBeenRounded	2	Indicates that the time has been rounded.
kCMTimeFlags_ImpliedValueFlagsMask	28	Indicates that the time is +infinity, -infinity, or indefinite. Use this value with bitwiseAnd on the flags.
kCMTimeFlags_Indefinite	16	Indicates that the time is indefinite.
kCMTimeFlags_NegativeInfinity	8	Indicates that the time is -infinity.
kCMTimeFlags_PositiveInfinity	4	Indicates that the time is +infinity.
kCMTimeFlags_Valid	1	Indicates that the time is valid.

Rounding Methods

Constant	Value	Description
kCMTimeRoundingMethod_Default	1	Synonym for kCMTimeRoundingMethod_RoundH
kCMTimeRoundingMethod_QuickTime	4	Use kCMTimeRoundingMethod_RoundTowardZer to smaller scale (that is, from more precision to les TimeRoundingMethod_RoundAwayFromZero if c larger scale (i.e. from less precision to more precisi Also, never round a negative number down to 0; magnitude negative CMTime in this case (-1/newT
kCMTimeRoundingMethod_RoundAwayFromZero	3	Round away from zero if abs(fraction) is >0.
kCMTimeRoundingMethod_RoundHalfAwayFromZero	1	Round towards zero if abs(fraction) is <0.5, away fr 0.5.
kCMTimeRoundingMethod_RoundTowardNegativeInfinity	6	Round towards -infinity if fraction is not 0.
kCMTimeRoundingMethod_RoundTowardPositiveInfinity	5	Round towards +infinity if fraction is not 0.
kCMTimeRoundingMethod_RoundTowardZero	2	Round towards zero if fraction is not 0.

4.105 class CMTimeRangeMBS

4.105.1 class CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The class that represent time ranges.
Example:

```
dim t as CMTimeMBS = CMTimeMBS.MakeWithSeconds(5) // start at 5
dim d as CMTimeMBS = CMTimeMBS.MakeWithSeconds(10) // duration 10
dim r as CMTimeRangeMBS = CMTimeRangeMBS.Make(t,d)
```

Notes: A CMTimeRange is represented as two CMTime structs, one that specifies the start time of the range and another that specifies the duration of the range. A time range does not include the end time that would be calculated by adding the duration to the start time.
Blog Entries

- Automated video editing with Xojo

4.105.2 Methods

4.105.3 AllTimeRange as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a special timerange to cover all times.
Notes: From zero to infinity.

4.105.4 Constructor(start as CMTimeMBS, duration as CMTimeMBS)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a new range with given values.
Example:

```
dim t as CMTimeMBS = new CMTimeMBS( 5*600, 600, kCMTimeFlags_Valid) // start at 5
dim d as CMTimeMBS = new CMTimeMBS(10*600, 600, kCMTimeFlags_Valid) // duration 10
dim r as CMTimeRangeMBS = new CMTimeRangeMBS(t,d)
```

4.105.5 ContainsTime(time as CMTimeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Indicates whether a time is contained within a time range.
Notes: time: CMTime to be tested for inclusion.
Returns true if the specified time is contained within the specified time range, false if it is not.

This function returns a Boolean value that indicates whether the time specified by the time parameter is contained within the range specified by the range parameter.

4.105.6 ContainsTimeRange(timeRange as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean that indicates whether a given time range is contained within another time range.
Notes: range1: The CMTimeRange being interrogated.
range2: CMTimeRange to be tested for inclusion.

Returns true if the second time range is contained within the first time range, false if it is not.
This function returns a Boolean value that indicates whether the time range specified by the range1 parameter contains the range specified by the range2 parameter.

4.105.7 Description as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a String with a description of a CMTimeRange.

4.105.8 Equal(range1 as CMTimeRangeMBS, range2 as CMTimeRangeMBS) as boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether two CMTimeRanges are identical.
Notes: range1: CMTimeRange to be compared for equality.
range2: Another CMTimeRange to be compared for equality.

Returns true if the two time ranges are identical, false if they differ.
This function returns a Boolean value that indicates whether the time ranges specified by the range1 and

range2 parameters are identical.

4.105.9 Intersection(range1 as CMTimeRangeMBS, range2 as CMTimeRangeMBS) as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the intersection of two CMTimeRanges.
Notes: range1: CMTimeRange to be intersected.
range2: Another CMTimeRange to be intersected.

Returns the intersection of the two CMTimeRanges.
This function returns a CMTimeRange structure that represents the intersection of the time ranges specified by the range1 and range2 parameters. This is the largest range that both ranges include.

4.105.10 kCMTimeRangeInvalid as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Use this constant to generate an invalid CMTimeRange.

4.105.11 kCMTimeRangeZero as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Use this constant to generate an empty CMTimeRange at 0.

4.105.12 Make(start as CMTimeMBS, duration as CMTimeMBS) as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a valid CMTimeRange with the given start time and duration.
Example:

```
dim t as CMTimeMBS = CMTimeMBS.MakeWithSeconds(5) // start at 5
dim d as CMTimeMBS = CMTimeMBS.MakeWithSeconds(10) // duration 10
dim r as CMTimeRangeMBS = CMTimeRangeMBS.Make(t,d)
```

Notes: start: CMTime for initializing the start field of the resulting CMTimeRange.

duration: CMTime for initializing the duration field of the resulting CMTimeRange.

Returns the resulting CMTimeRange.
The duration parameter must have an epoch of 0; otherwise an invalid time range will be returned.

4.105.13 Show

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Prints a description of the CMTimeRange to stderr (similar to CFShow).

4.105.14 TimeRangeFromTimeToTime(start as CMTimeMBS, EndTime as CMTimeMBS) as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a valid CMTimeRange from the given start and end time.
Notes: start: CMTime structure representing start time for creating the range.
EndTime: CMTime structure representing end time for creating the range.
Returns the resulting CMTimeRange.

4.105.15 Union(range1 as CMTimeRangeMBS, range2 as CMTimeRangeMBS) as CMTimeRangeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the union of two CMTimeRanges.
Notes: range1: CMTimeRange to be unified.
range2: Another CMTimeRange to be unified.

Returns the union of the two CMTimeRanges.

This function returns a CMTimeRange that represents the union of the time ranges specified by the range1 and range2 parameters. This is the smallest range that includes all times that are in either range.

4.105.16 Properties

4.105.17 Duration as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The duration.
Notes: (Read and Write property)

4.105.18 EndTime as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a CMTime structure representing the end of a time range.
Notes: range: The CMTimeRange from which to find the end of time range.
Returns a CMTime representing the end of the specified time range.

This function returns a CMTime structure that indicates the end of the time range specified by the range parameter. ContainsTime(range, range.EndTime) is always false.
(Read only property)

4.105.19 IsEmpty as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given CMTimeRange has a duration of 0.
Notes: Returns true if range has a duration of 0; otherwise, false.
(Read only property)

4.105.20 IsIndefinite as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given CMTimeRange is indefinite.
Notes: Returns true if range is indefinite; otherwise, false.
An indefinite time range has either an indefinite start or an indefinite duration, or both.
(Read only property)

4.105.21 IsInvalid as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given CMTimeRange is invalid.
Notes: Returns true if range is invalid; otherwise, false.
(Read only property)

4.105.22 IsValid as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given CMTimeRange is valid.
Notes: Returns true if range is valid; otherwise, false.
(Read only property)

4.105.23 Start as CMTimeMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The start time.
Notes: (Read and Write property)

4.106 class CVImageBufferMBS

4.106.1 class CVImageBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Core Video image buffers provides a convenient interface for managing different types of image data.
Example:

```
dim p as new Picture(300, 200)
dim b as CVPixelBufferMBS = CVPixelBufferMBS.PixelBufferWithPicture(p)
MsgBox str(b.Width)+" x "+str(b.Height)
```

Notes: Pixel buffers and Core Video OpenGL buffers derive from the Core Video image buffer.
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 21.5pr3
- MBS Xojo Plugins, version 21.2pr1
- MBS Xojo Plugins, version 18.4pr3
- AVCaptureVideoDataOutput improvements
- MBS Xojo / Real Studio Plugins, version 15.2pr8

4.106.2 Methods

4.106.3 CIImage as Variant

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The CIImage object for drawing into picture.
Notes: Each time you call this function a new CIImage object is created, so cache it if possible.

4.106.4 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The private constructor.

4.106.5 JPEG(CompressionFactor as Double = 0.8) as Memoryblock

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns JPEG compressed image data.
Notes: Compression Factor is in range from 0 to 1.0.
Returns memoryblock with jpeg data or nil in case of error.

4.106.6 NSImage as Variant

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Queries image buffer as NSImageMBS object.
Notes: Result is NSImageMBS object or nil in case of error.

4.106.7 Properties

4.106.8 Context as CGContextMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The CGContext for this picture.
Notes: (Read only property)

4.106.9 Handle as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

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

4.106.10 IsFlipped as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns whether the image is flipped vertically or not.
Notes: True if 0,0 in the texture is upper left, false if 0,0 is lower left.
(Read only property)

4.106.11 Lasterror as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The last error.
Notes: (Read and Write property)

4.106.12 Picture as Picture

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: The picture this image buffer references to.
Notes: (Read only property)

4.107 class CVPixelBufferMBS

4.107.1 class CVPixelBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: A Core Video pixel buffer is an image buffer that holds pixels in main memory.
Example:

```
dim p as new Picture(300, 200)
dim b as CVPixelBufferMBS = CVPixelBufferMBS.PixelBufferWithPicture(p)
MsgBox str(b.Width)+" x "+str(b.Height)
```

Notes: Applications generating frames, compressing or decompressing video, or using Core Image can all make use of Core Video pixel buffers.
Subclass of the CVImageBufferMBS class.
Blog Entries

- MBS Xojo Plugins, version 22.3pr4
- MBS Xojo Plugins, version 21.5pr3
- MBS Xojo Plugins, version 21.2pr1
- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.4
- MBS Xojo Plugins, version 19.4pr4
- MBS Xojo Plugins, version 19.4pr2
- MBS Xojo / Real Studio Plugins, version 14.5pr4

Xojo Developer Magazine

- 12.2, page 47: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz

4.107.2 Methods

4.107.3 BaseAddressOfPlane(planeIndex as Integer) as Ptr

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the base address of the plane at the specified plane index.
Notes: planeIndex: The index of the plane.

Returns the base address of the plane, or NULL for nonplanar pixel buffers.
Retrieving the base address for a pixel buffer requires that the buffer base address be locked by a successful

call to LockBaseAddress.

4.107.4 BytesPerRowOfPlane(planeIndex as Integer) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the number of bytes per row for a plane at the specified index in the pixel buffer.
Notes: planeIndex: The index of the plane whose bytes-per-row value you want to obtain.

Returns the number of row bytes of the plane, or nil for nonplanar pixel buffers.

4.107.5 CIImage as Variant

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates CIImageMBS object for given pixel buffer.

4.107.6 Constructor(pic as picture)

Plugin Version: 13.2, Platforms: macOS, iOS, Targets: All.

Function: Creates a pixel buffer for a picture.
Example:

```
// Let's test the class:

// get a picture
Dim p As Picture = LogoMBS(500)

// load into CVPixelBufferMBS
Dim c As New CVPixelBufferMBS(p)

// ask back as JPEG
Dim JPEGData As MemoryBlock = c.JPEG(0.8)

// show it
ImageViewer1.Image = Picture.FromData(JPEGData)
```

Notes: Only for Cocoa target.
This function takes the existing picture and creates a pixelbuffer referencing it. Also we setup the CGContext object and stores it in Context property. Also picture property is set to the current picture.

Implemented for iOS in version 22.3 of MBS Plugin.
See also:

- 4.107.7 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 708
- 4.107.8 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.9 Constructor(Width as Integer, Height as Integer, PixelFormat as String, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.10 Constructor(Width as Integer, Height as Integer, PixelFormat as String, pixelBufferAttributes as Dictionary) 710

4.107.7 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary)

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates a pixel buffer for a given size and pixel format containing data specified by a memory location.
Notes: width: The width of the pixel buffer, in pixels.
height: The height of the pixel buffer, in pixels.
pixelFormatType: The pixel format identified by its respective four character code
baseAddress: The memoryblock referenced by the PixelBuffer with the data.
bytesPerRow: The row bytes of the pixel storage memory.
pixelBufferAttributes: A Core Foundation dictionary with additional attributes for a pixel buffer. This parameter is optional.

Returns a Core Video result code. See Core Video Constants for possible values.
Some of the parameters specified in this call override equivalent pixel buffer attributes. For example, if you define the kCVPixelBufferWidth and kCVPixelBufferHeight keys in the pixel buffer attributes parameter (pixelBufferAttributes), these values are overridden by the width and height parameters.
See also:

- 4.107.6 Constructor(pic as picture) 707
- 4.107.8 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.9 Constructor(Width as Integer, Height as Integer, PixelFormat as String, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.10 Constructor(Width as Integer, Height as Integer, PixelFormat as String, pixelBufferAttributes as Dictionary) 710

4.107.8 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, pixelBufferAttributes as Dictionary)

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates a single pixel buffer for a given size and pixel format.
Notes: width: Width of the pixel buffer, in pixels.
height: Height of the pixel buffer, in pixels.
pixelFormatType: The pixel format identified by its respective four-character code.
pixelBufferAttributes: A dictionary with additional attributes for a pixel buffer. This parameter is optional.

Raises exception on failure with error code included.

This function allocates the necessary memory based on the pixel dimensions, format, and extended pixels described in the pixel buffer,Äôs attributes.

Some of the parameters specified in this call override equivalent pixel buffer attributes. For example, if you define the kCVPixelBufferWidth and kCVPixelBufferHeight keys in the pixel buffer attributes parameter (pixelBufferAttributes), these values are overridden by the width and height parameters.
See also:

- 4.107.6 Constructor(pic as picture) 707
- 4.107.7 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 708
- 4.107.9 Constructor(Width as Integer, Height as Integer, PixelFormat as String, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.10 Constructor(Width as Integer, Height as Integer, PixelFormat as String, pixelBufferAttributes as Dictionary) 710

4.107.9 Constructor(Width as Integer, Height as Integer, PixelFormat as String, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary)

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates a pixel buffer for a given size and pixel format containing data specified by a memory location.
Notes: width: The width of the pixel buffer, in pixels.
height: The height of the pixel buffer, in pixels.
pixelFormatType: The pixel format identified by its respective four character code
baseAddress: The memoryblock referenced by the PixelBuffer with the data.
bytesPerRow: The row bytes of the pixel storage memory.
pixelBufferAttributes: A Core Foundation dictionary with additional attributes for a pixel buffer. This

parameter is optional.

Returns a Core Video result code. See Core Video Constants for possible values.
Some of the parameters specified in this call override equivalent pixel buffer attributes. For example, if you define the kCVPixelBufferWidth and kCVPixelBufferHeight keys in the pixel buffer attributes parameter (pixelBufferAttributes), these values are overridden by the width and height parameters.
See also:

- 4.107.6 Constructor(pic as picture) 707
- 4.107.7 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 708
- 4.107.8 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, pixelBufferAttributes as Dictionary) 709
- 4.107.10 Constructor(Width as Integer, Height as Integer, PixelFormat as String, pixelBufferAttributes as Dictionary) 710

4.107.10 Constructor(Width as Integer, Height as Integer, PixelFormat as String, pixelBufferAttributes as Dictionary)

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates a single pixel buffer for a given size and pixel format.
Notes: width: Width of the pixel buffer, in pixels.
height: Height of the pixel buffer, in pixels.
pixelFormatType: The pixel format identified by its respective four-character code.
pixelBufferAttributes: A dictionary with additional attributes for a pixel buffer. This parameter is optional.

Raises exception on failure with error code included.

This function allocates the necessary memory based on the pixel dimensions, format, and extended pixels described in the pixel buffer,Äôs attributes.

Some of the parameters specified in this call override equivalent pixel buffer attributes. For example, if you define the kCVPixelBufferWidth and kCVPixelBufferHeight keys in the pixel buffer attributes parameter (pixelBufferAttributes), these values are overridden by the width and height parameters.
See also:

- 4.107.6 Constructor(pic as picture) 707
- 4.107.7 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 708
- 4.107.8 Constructor(Width as Integer, Height as Integer, PixelFormat as Integer, pixelBufferAttributes as Dictionary) 709

- 4.107.9 Constructor(Width as Integer, Height as Integer, PixelFormat as String, data as MemoryBlock, bytesPerRow as Integer, pixelBufferAttributes as Dictionary) 709

4.107.11 FillExtendedPixels

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Fills the extended pixels of the pixel buffer.
Notes: This function replicates edge pixels to fill the entire extended region of the image.

4.107.12 Flush

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Flushes output to the image buffer.
Notes: Anything not yet drawn is drawn now.

4.107.13 GetExtendedPixels(byref extraColumnsOnLeft as Integer, byref extraColumnsOnRight as Integer, byref extraRowsOnTop as Integer, byref extraRowsOnBottom as Integer)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the amount of extended pixel padding in the pixel buffer.
Notes: extraColumnsOnLeft: On output, the pixel row padding to the left.
extraColumnsOnRight: On output, the pixel row padding to the right.
extraRowsOnTop: On output, the pixel row padding to the top.
extraRowsOnBottom: On output, the pixel row padding to the bottom.

4.107.14 HeightOfPlane(planeIndex as Integer) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the height of the plane at planeIndex in the pixel buffer.
Notes: planeIndex: The index of the plane.
Returns the height of the buffer, in pixels, or 0 for nonplanar pixel buffers.

4.107.15 kCVPixelBufferBytesPerRowAlignmentKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Indicates the number of bytes per row in the pixel buffer (type Number).

4.107.16 kCVPixelBufferCGBitmapContextCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Indicates whether the pixel buffer is compatible with Core Graphics bitmap contexts (type Boolean).

4.107.17 kCVPixelBufferCGImageCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Indicates whether the pixel buffer is compatible with CGImage types (type Boolean).

4.107.18 kCVPixelBufferExtendedPixelsBottomKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The number of pixels padding the bottom of the image (type Number).

4.107.19 kCVPixelBufferExtendedPixelsLeftKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The number of pixels padding the left of the image (type Number).

4.107.20 kCVPixelBufferExtendedPixelsRightKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The number of pixels padding the right of the image (type Number).

4.107.21 kCVPixelBufferExtendedPixelsTopKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The number of pixels padding the top of the image (type Number).

4.107.22 kCVPixelBufferHeightKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The height of the pixel buffer (type Number).

4.107.23 kCVPixelBufferIOSurfaceCoreAnimationCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: A boolean value

4.107.24 kCVPixelBufferIOSurfaceOpenGLFBOCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Ensures that the CVPixelBuffer's IOSurfaceRef can be displayed in an CoreAnimation CALayer.

4.107.25 kCVPixelBufferIOSurfaceOpenGLTextureCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Ensures that CGLTexImageIOSurface2D() will succeed in creating a valid texture object from the CVPixelBuffer's IOSurface AND that the resulting texture may be used as a color buffer attachment to a OpenGL frame buffer object.

4.107.26 kCVPixelBufferIOSurfacePropertiesKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Indicates optional properties in the IOSurface framework (type Dictionary). An empty dictionary indicates default values.
Presence of this key requests allocation via the IOSurface framework.

4.107.27 kCVPixelBufferMemoryAllocatorKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The allocator used with this buffer (type CFAllocatorRef).

4.107.28 kCVPixelBufferOpenGLCompatibilityKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Indicates whether the pixel buffer is compatible with Core Graphics bitmap contexts (type Boolean).

4.107.29 kCVPixelBufferPixelFormatTypeKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The pixel format for this buffer (type Number, or type Array containing an array of Number types (actually type OSType)). For a listing of common pixel formats, see the QuickTime Ice Floe Dispatch 20.

4.107.30 kCVPixelBufferPlaneAlignmentKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: Specifies the alignment of the planes within the buffer. Planes will start on a byte number which is a multiple of this value. (type Number).

4.107.31 kCVPixelBufferWidthKey as string

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: One of the attribute keys.
Notes: The width of the pixel buffer (type Number).

4.107.32 LockBaseAddress(flags as Integer = 0)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Locks the base address of the pixel buffer.
Notes: lockFlags: You can pass zero or kLockReadonly.

4.107.33 PixelBufferWithCGImage(CGImage as Variant) as CVPixelBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a pixel buffer for a CGImage by making a copy of the picture.
Notes: Image parameter must be a CGImageMBS object.
For Carbon and Cocoa.
Returns nil on any error.

4.107.34 PixelBufferWithHandle(Handle as Integer) as CVPixelBufferMBS

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Creates pixel buffer object with handle.
Notes: Handle must be a CVPixelBufferRef, which we retain.

4.107.35 PixelBufferWithPicture(pic as picture) as CVPixelBufferMBS

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Creates a pixel buffer for a picture by making a copy of the picture.
Example:

```
dim p as new Picture(300, 200)
dim b as CVPixelBufferMBS = CVPixelBufferMBS.PixelBufferWithPicture(p)
MsgBox str(b.Width)+" x "+str(b.Height)
```

Notes: For Carbon and Cocoa.
Returns nil on any error.

4.107.36 PixelData as MemoryBlock

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Copies pixel data into MemoryBlock.
Notes: Returns nil in case of error.
Works only on non planar pixel buffers.

4.107.37 SetPixelData(data as MemoryBlock) as Boolean

Plugin Version: 19.4, Platform: macOS, Targets: All.

Function: Copies pixel data from MemoryBlock into pixel buffer.
Notes: Returns false in case of error and true for success.
Works only on non planar pixel buffers.

4.107.38 UnlockBaseAddress(flags as Integer = 0)

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Unlocks the base address of the pixel buffer.

4.107.39 WidthOfPlane(planeIndex as Integer) as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the width of the plane at a given index in the pixel buffer.
Notes: pixelBuffer: The pixel buffer whose plane width you want to obtain.
planeIndex: The plane index that contains the plane's width value.

Returns the width of the plane, in pixels, or 0 for nonplanar pixel buffers.

4.107.40 Properties

4.107.41 BaseAddress as Ptr

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the base address of the pixel buffer.
Notes: For chunky buffers, returns a pointer to the pixel at (0,0) in the buffer.

For planar buffers, returns a pointer to a PlanarComponentInfo structure (as defined by QuickTime in ImageCodec.h).

Retrieving the base address for a pixel buffer requires that the buffer base address be locked via a successful call to CVPixelBufferLockBaseAddress.
(Read only property)

4.107.42 BytesPerRow as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the number of bytes per row of the pixel buffer.
Example:

```
dim p as new Picture(300, 200)
dim b as new CVPixelBufferMBS(p)
MsgBox str(b.BytesPerRow)
```

Notes: The number of bytes per row of the image data. For planar buffers, this function returns a rowBytes value such that bytesPerRow * height covers the entire image, including all planes.
(Read only property)

4.107.43 DataSize as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the data size for contiguous planes of the pixel buffer.
Example:

```
dim p as new Picture(300, 200)
dim b as new CVPixelBufferMBS(p)
MsgBox str(b.DataSize)
```

Notes: (Read only property)

4.107.44 Height as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the height of the pixel buffer.

Example:

```
dim p as new Picture(300, 200)
dim b as CVPixelBufferMBS = CVPixelBufferMBS.PixelBufferWithPicture(p)
MsgBox str(b.Width)+" x "+str(b.Height)
```

Notes: (Read only property)

4.107.45 IsPlanar as Boolean

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Determines whether the pixel buffer is planar.
Notes: Returns true if the pixel buffer was created using CVPixelBufferCreateWithPlanarBytes; otherwise, false.
(Read only property)

4.107.46 PixelFormatType as String

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the pixel format type of the pixel buffer.
Notes: Returns a four-character code OSType identifier for the pixel format.
(Read only property)

4.107.47 PlaneCount as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns number of planes of the pixel buffer.
Notes: Returns the number of planes. Returns 0 for nonplanar pixel buffers.
(Read only property)

4.107.48 Width as Integer

Plugin Version: 13.2, Platform: macOS, Targets: All.

Function: Returns the width of the pixel buffer.
Example:

```
dim p as new Picture(300, 200)
dim b as CVPixelBufferMBS = CVPixelBufferMBS.PixelBufferWithPicture(p)
MsgBox str(b.Width)+" x "+str(b.Height)
```

Notes: (Read only property)

4.107.49 Constants

Constants

Constant	Value	Description
kLockReadonly	1	The flag for LockBaseAddress function. If you are not going to modify the data while you hold the lock, you should set this flag to avoid potentially invalidating any existing caches of the buffer contents. This flag should be passed both to the lock and unlock functions. Non-symmetrical usage of this flag will result in undefined behavior.

Pixel Formats

Constant	Value	Description
kCVPixelFormatType_128RGBAFloat	”RGfA”	128 bit RGBA IEEE float, 32-bit little-endian
kCVPixelFormatType_14Bayer_BGGR	”bgg4”	Bayer 14-bit Little-Endian, packed in 16-bits, with G R G R...
kCVPixelFormatType_14Bayer_GBRG	”gbr4”	Bayer 14-bit Little-Endian, packed in 16-bits, with R G R G...
kCVPixelFormatType_14Bayer_GRBG	”grb4”	Bayer 14-bit Little-Endian, packed in 16-bits, with B G B G...
kCVPixelFormatType_14Bayer_RGGB	”rgg4”	Bayer 14-bit Little-Endian, packed in 16-bits, with G B G B...
kCVPixelFormatType_16BE555	&h00000010	16 bit BE RGB 555. Available in OS X v10.5 and later.
kCVPixelFormatType_16BE565	”B565”	16 bit BE RGB 565. Available in OS X v10.5 and later.
kCVPixelFormatType_16Gray	”b16g”	16 bit Grayscale, 16-bit big-endian samples, b Available in OS X v10.5 and later.
kCVPixelFormatType_16LE555	”L555”	16 bit LE RGB 555. Available in OS X v10.5 and later.
kCVPixelFormatType_16LE5551	”5551”	16 bit LE RGB 5551. Available in OS X v10.5 and later.
kCVPixelFormatType_16LE565	”L565”	16 bit LE RGB 565. Available in OS X v10.5 and later.
kCVPixelFormatType_1IndexedGray_WhiteIsZero	&h00000021	1 bit indexed gray, white is zero. Available in OS X v10.5 and later.
kCVPixelFormatType_1Monochrome	&h00000001	1 bit indexed. Available in OS X v10.5 and later.
kCVPixelFormatType_24BGR	”24BG”	24 bit BGR. Available in OS X v10.5 and later.
kCVPixelFormatType_24RGB	&h00000018	24 bit RGB. Available in OS X v10.5 and later.
kCVPixelFormatType_2Indexed	&h00000002	2 bit indexed. Available in OS X v10.5 and later.
kCVPixelFormatType_2IndexedGray_WhiteIsZero	&h00000022	2 bit indexed gray, white is zero. Available in OS X v10.5 and later.
kCVPixelFormatType_30RGB	”R10k”	30 bit RGB, 10-bit big-endian samples, 2 unu cant end).
kCVPixelFormatType_30RGBLEPackedWideGamut	”w30r”	little-endian RGB101010, 2 MSB are zero, wi
kCVPixelFormatType_32ABGR	”ABGR”	32 bit ABGR. Available in OS X v10.5 and later.
kCVPixelFormatType_32AlphaGray	”b32a”	32 bit AlphaGray, 16-bit big-endian samples, Available in OS X v10.5 and later.
kCVPixelFormatType_32ARGB	&h00000020	32 bit ARGB. Available in OS X v10.5 and later.
kCVPixelFormatType_32BGRA	”BGRA”	32 bit BGRA. Available in OS X v10.5 and later.
kCVPixelFormatType_32RGBA	”RGBA”	32 bit RGBA. Available in OS X v10.5 and later.
kCVPixelFormatType_420YpCbCr10BiPlanarFullRange	”xf20”	2 plane YCbCr10 4:2:0, each 10 bits in the MS 0-1023)
kCVPixelFormatType_420YpCbCr10BiPlanarVideoRange	”x420”	2 plane YCbCr10 4:2:0, each 10 bits in the MS [64,940] chroma= [64,960])
kCVPixelFormatType_420YpCbCr8BiPlanarFullRange	”420f”	Bi-Planar Component Y’CbCr 8-bit 4:2:0, full [1,255]). baseAddr points to a big-endian CV BiPlanar structure.
kCVPixelFormatType_420YpCbCr8BiPlanarVideoRange	”420v”	Bi-Planar Component Y’CbCr 8-bit 4:2:0, chroma= [16,240]). baseAddr points to a b Info_YCbCrBiPlanar structure.
kCVPixelFormatType_420YpCbCr8Planar	”y420”	Planar Component Y’CbCr 8-bit 4:2:0. baseA PlanarPixelBufferInfo_YCbCrPlanar struct.

4.108 control DesktopAVCaptureViewControlMBS

4.108.1 control DesktopAVCaptureViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The capture control for Mac.
Notes: AVCaptureView is a subclass of NSView that can be used to display standard user interface controls for capturing media data.
MBS Plugin wraps this as a control for Xojo.
Please notice this is 64-bit only and requires MacOS 10.9 or newer.
Blog Entries

- MBS Xojo Plugins, version 22.0pr8
- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- AVCaptureVideoDataOutput improvements
- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr2

Videos

- Presentation from Munich conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 16.5, page 9: News

4.108.2 Methods

4.108.3 Available as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

4.108.4 setSession(session as AVCaptureSessionMBS, showVideoPreview as boolean, showAudioPreview as boolean)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Sets the session represented by this view.
Notes: session: The session to be represented.
showVideoPreview: Whether or not video preview should be shown. If YES, capture inputs for video media data will be added, removed, or modified depending on device availability and user selection.
showAudioPreview: Whether or not audio preview should be shown. If YES, capture inputs for audio media data will be added, removed, or modified depending on device availability and user selection.

The view must either show audio preview or video preview or both. Furthermore, the view may modify the capture session, for example, to access media data for preview or when the user select a new capture source. Only the default session is started and stopped automatically. The provided session must be manually started and stopped.

4.108.5 Properties

4.108.6 controlsStyle as Integer

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The style of the capture controls pane associated with the view.
Notes: (Read and Write property)

4.108.7 fileOutput as AVCaptureFileOutputMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A capture file output used to record media data.
Notes: The value of this property is the first instance of AVCaptureFileOutput contained in the session's outputs array or nil if no such instance is found. In the latter case the capture view's start recording button will be disabled. However, the controls for choosing input sources may still be enabled.
(Read only property)

4.108.8 session as AVCaptureSessionMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A capture session represented by this view.
Notes: Modifying the capture session will impact its visual representation in the view. The default value is

a session configured for movie file recordings of audio and video media data. Use setSession() to change the value of this property.
(Read only property)

4.108.9 videoGravity as String

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A string defining how the video is displayed within the views bounds rect.
Notes: Options are AVLayerVideoGravityResize, AVLayerVideoGravityResizeAspect and AVLayerVideoGravityResizeAspectFill. AVLayerVideoGravityResizeAspect is default. See <AVFoundation/AVAnimation.h>for a description of these options.
(Read and Write property)

4.108.10 View as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The reference to the AVCaptureView object.
Notes: (Read only property)

4.108.11 Events

4.108.12 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.108.13 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.

4.108.14 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.

4.108.15 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.

4.108.16 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.

4.108.17 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.

4.108.18 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.

4.108.19 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.

4.108.20 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.

4.108.21 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.

4.108.22 startRecordingToFileOutput(fileOutput as AVCaptureFileOutputMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Informs you that a new media recording should be started.
Notes: If captureFileOutput is an instance of AVCaptureMovieFileOutput this can be achieved by calling

startRecordingToOutputFileURL on the captureFileOutput.

4.108.23 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.

4.108.24 Constants

Constants Groups

Constant	Value	Description
ControlsStyleDefault	0	The default controls pane is associated with the view.
ControlsStyleFloating	1	The floating controls pane for media recordings is associated with the view.
ControlsStyleInline	0	The inline controls pane for media recording is associated with the view.
ControlsStyleInlineDeviceSelection	2	The inline controls pane for selection capture devices is associated with tl view.

4.109 control DesktopAVPlayerViewControlMBS

4.109.1 control DesktopAVPlayerViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The player control for Mac.
Notes: AVPlayerView is a subclass of NSView that can be used to display the visual content of an AVPlayer object and the standard playback controls.
MBS Plugin wraps this as a control for Xojo.
Please notice this is 64-bit only and requires MacOS 10.9 or newer.
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
- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr2

Videos

- Presentation from Munich conference about MBS Plugins.
- Presentation from Xojo Developer Conference 2019 in Miami.

4.109.2 Methods

4.109.3 Available as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.

4.109.4 beginTrimming

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Sets the controls panel into trimming mode and blocks until the user selects either the Trim or the Cancel button.
Notes: Triggers TrimmingCompleted later.

4.109.5 flashChapterNumber(chapterNumber as Integer, chapterTitle as String)

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Display the provided chapter number and title momentarily.
Notes: chapterNumber: The chapter number (required).
chapterTitle: The chapter title (optional).

4.109.6 Properties

4.109.7 actionPopUpButtonMenu as NSMenuMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The action popup menu.
Notes: Clients can set this property in order to show an action pop up button. Default is nil.
(Read and Write property)

4.109.8 allowsPictureInPicturePlayback as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the receiver allows Picture in Picture playback.
Notes: Default is false.
Available in MacOS 10.15 or later.
(Read and Write property)

4.109.9 canBeginTrimming as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the current media can be trimmed.
Notes: (Read only property)

4.109.10 contentOverlayView as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Use the content overlay view to add additional custom views between the video content and the controls.

Notes: (Read only property)

4.109.11 controlsStyle as Integer

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The style of the playback controls pane currently associated with the view.
Notes: (Read and Write property)

4.109.12 player as AVPlayerMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The player from which to source the media content for the view.
Notes: (Read and Write property)

4.109.13 readyForDisplay as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Boolean indicating that the first video frame has been made ready for display for the current item of the associated AVPlayer.
Notes: (Read only property)

4.109.14 showsFrameSteppingButtons as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Replace scanning controls in the playback UI with frame stepping buttons.
Notes: Default is false.
(Read and Write property)

4.109.15 showsFullScreenToggleButton as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the controls pane will show a full screen toggle button.
Notes: Default is false.
(Read and Write property)

4.109.16 showsSharingServiceButton as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the controls pane will show a sharing service button when the current player item can be shared.
Notes: Default is false.
(Read and Write property)

4.109.17 showsTimecodes as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: If timecodes are available, allow the AVPlayerView controls to enter timecode mode.
Notes: Default is false.
Available in MacOS 10.15 or later.
(Read and Write property)

4.109.18 updatesNowPlayingInfoCenter as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the now playing info center should be updated.
Notes: Default is true.
(Read and Write property)

4.109.19 videoBounds as NSRectMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The current size and position of the video image as displayed within the receiver's view's bounds.
Notes: (Read only property)

4.109.20 videoGravity as String

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: A string defining how the video is displayed within an AVPlayerLayer bounds rect.
Notes: Options are AVLayerVideoGravityResizeAspect, AVLayerVideoGravityResizeAspectFill and AVLayerVideoGravityResize. AVLayerVideoGravityResizeAspect is default.
See <AVFoundation/AVAnimation.h>for a description of these options.

(Read and Write property)

4.109.21 View as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The reference to the AVPlayerView object.
Notes: (Read only property)

4.109.22 Events

4.109.23 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.109.24 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.

4.109.25 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.

4.109.26 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.

4.109.27 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.

4.109.28 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.

4.109.29 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.

4.109.30 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.

4.109.31 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.

4.109.32 playerViewDidStartPictureInPicture

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture did start.

4.109.33 playerViewDidStopPictureInPicture

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture did stop.

4.109.34 playerViewFailedToStartPictureInPicture(error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture failed to start.
Notes: error: An error describing why it failed.

4.109.35 playerViewRestoreUserInterfaceForPictureInPictureStop as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to restore the user interface before Picture in Picture stops.
Notes: Please return true if you restored it or false if not, so we can call completion handler passing the value.

Default is to pass true if you don't implement this event.

4.109.36 playerViewShouldAutomaticallyDismissAtPictureInPictureStart as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this event and return false to prevent player view from automatically being miniaturized or losing focus when Picture in Picture starts.
Notes: If you implement this, please return true or false.
By default we return false if you don't

4.109.37 playerViewWillStartPictureInPicture

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture will start.

4.109.38 playerViewWillStopPictureInPicture

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Control can implement this method to be notified when Picture in Picture will stop.

4.109.39 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.

4.109.40 TrimmingCompleted(result as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event handler that is executed when the user selects either the Trim or Cancel button in the trimming UI.

4.109.41 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.

4.109.42 Constants

Control Styles

Constant	Value	Description
ControlsStyleDefault	1	The default controls pane is associated with the view.
ControlsStyleFloating	2	The floating controls pane is associated with the view.
ControlsStyleInline	1	The inline controls pane is associated with the view.
ControlsStyleMinimal	3	The minimal controls pane is associated with the view.
ControlsStyleNone	0	No controls pane is associated with the view.

Trim Results

Constant	Value	Description
TrimCancelButton	1	The user selected the Cancel button.
TrimOKButton	0	The user selected the Trim button.

4.110 control DesktopAVRoutePickerViewControlMBS

4.110.1 control DesktopAVRoutePickerViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: An object that displays controls for picking playback routes.
Notes: Available on MacOS 10.15 or newer.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- MBS Xojo Plugins, version 20.1pr2
- Airplay related classes for AVFoundation

Videos

- XDC 2020 MBS Plugins Presentation

Xojo Developer Magazine

- 18.3, page 10: News

4.110.2 Properties

4.110.3 player as AVPlayerMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The player for which to perform routing operations.
Notes: (Read and Write property)

4.110.4 routePickerButtonBordered as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether or not the picker button has a border.
Notes: Default is true.
(Read and Write property)

4.110.5 View as NSViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The view object.
Notes: (Read only property)

4.110.6 Available as Boolean

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Whether this control is available.
Notes: (Read and Write computed property)

4.110.7 routePickerButtonColorForState(state as integer) as NSColorMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Defines the color for the button for a given state.
Notes: (Read and Write computed property)

4.110.8 Events

4.110.9 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

4.110.10 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.

4.110.11 DidEndPresentingRoutes

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells you that the route picker view has finished presenting routes to the user.

4.110.12 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.

4.110.13 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.

4.110.14 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.

4.110.15 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.

4.110.16 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.

4.110.17 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.

4.110.18 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.

4.110.19 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.

4.110.20 WillBeginPresentingRoutes

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Tells you that the route picker view will start presenting routes to the user.

4.110.21 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.

4.110.22 Constants

Button States

Constant	Value	Description
StateActive	2	Active state of the picker. The picker has this state when AirPlay is active.
StateActiveHighlighted	3	Highlighted state of the active picker. The picker has this state when it is highlighted and AirPlay is active.
StateNormal	0	Normal or default state of the picker.
StateNormalHighlighted	1	Highlighted state of the picker. The picker has this state when a mouse-down event occurs inside the button. It loses this highlight when a mouse-up event occurs.

4.111 class DesktopMoviePlayer

4.111.1 class DesktopMoviePlayer

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Extends the Movieplayer class inside Xojo.

4.111.2 Methods

4.111.3 AVAssetMBS as AVAssetMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Returns the AVAsset for this movieplayer.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("Der Hausbau.m4v")
dim m as movie = f.OpenAsMovie

MoviePlayer1.movie = m
MoviePlayer1.play

dim asset as AVAssetMBS = MoviePlayer1.AVAssetMBS
MsgBox str(asset.duration.Seconds)+" seconds"
```

Notes: Useful in order to use plugin functions on a movieplayer, e.g. query meta data.

4.111.4 AVPlayerLayerMBS as AVPlayerLayerMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Returns the AVPlayerLayer for this movieplayer.
Notes: Only for Cocoa target in Xojo.

4.111.5 AVPlayerMBS as AVPlayerMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Returns the AVPlayer for this movieplayer.
Notes: Only for Cocoa target.

4.112 class Movie

4.112.1 class Movie

Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends the Movie class inside Xojo.

4.112.2 Methods

4.112.3 AVAssetMBS as AVAssetMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the AVAsset for this movie.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as movie = f.OpenAsMovie
dim asset as AVAssetMBS = m.AVAssetMBS
MsgBox str(asset.duration.Seconds)+" seconds"
```

Notes: Useful in order to use plugin functions on a movie object, e.g. query meta data.
Only for Cocoa target in Xojo 2014.

4.113 class MoviePlayer

4.113.1 class MoviePlayer

Platform: macOS, Targets: Desktop only.

Function: Extends the Movieplayer class inside Xojo.

4.113.2 Methods

4.113.3 AVAssetMBS as AVAssetMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the AVAsset for this movieplayer.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("Der Hausbau.m4v")
dim m as movie = f.OpenAsMovie

MoviePlayer1.movie = m
MoviePlayer1.play

dim asset as AVAssetMBS = MoviePlayer1.AVAssetMBS
MsgBox str(asset.duration.Seconds)+" seconds"
```

Notes: Useful in order to use plugin functions on a movieplayer, e.g. query meta data.
Only for Cocoa target.

4.113.4 AVPlayerLayerMBS as AVPlayerLayerMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the AVPlayerLayer for this movieplayer.
Notes: Only for Cocoa target in Xojo.

4.113.5 AVPlayerMBS as AVPlayerMBS

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the AVPlayer for this movieplayer.

Notes: Only for Cocoa target.

4.114 class SDAVAssetExportSessionMBS

4.114.1 class SDAVAssetExportSessionMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The SDAVAssetExportSession class for media export.
Notes: Based on open source SDAVAssetExportSession class:
https://github.com/rs/SDAVAssetExportSession

AVAssetExportSessionMBS drop-in replacement with customizable audio & video settings.

An SDAVAssetExportSessionMBS object transcodes the contents of an AVAsset source object to create an output of the form described by a specified audio and video settings.

After you have initialized an export session with the asset that contains the source media, and the output file type (outputFileType), and various other settings you can start the export running by invoking exportAsynchronously. Because the export is performed asynchronously, this method returns immediately—you can use progress to check on the progress. Depending on the capabilities of the device, some exports may be queued when multiple exports are attempted. When this happens, the status of a queued export will indicate that it's waiting (StatusWaiting).

The exportAsynchronouslyCompleted event is called whether the export fails, completes, or is cancelled. Upon completion, the status property indicates whether the export has completed successfully. If it has failed, the value of the error property supplies additional information about the reason for the failure.
Available in OS X v10.9 and later in 64-bit.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.3

- MBS Xojo Plugins, version 20.3pr5

4.114.2 Methods

4.114.3 available as Boolean

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Whether this class is available.
Notes: Returns true on Mac OS X 10.7 and newer in 64-bit applications.

4.114.4 cancelExport

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Cancels the execution of an export session.

4.114.5 Constructor(asset as AVAssetMBS)

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Initializes an asset export session with a specified asset and preset.
Notes: asset: The asset you want to export.

4.114.6 exportAsynchronously

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Starts the asynchronous execution of an export session.
Notes: This method starts an asynchronous export operation and returns immediately. status signals the terminal state of the export session, and if a failure occurs, error describes the problem.

This method calls AVFoundationMBS.exportAsynchronouslyCompleted event later. Tag is passed so you can pass information needed to finish work.

If internal preparation for export fails, event is invoked synchronously. The event may also be called asynchronously, after the method returns, in the following cases:

- If a failure occurs during the export, including failures of loading, re-encoding, or writing media data to the output.
- If cancelExport is invoked.
- After the export session succeeds, having completely written its output to the outputURL.

With tag you can pass any value you like to the event later. This can be for example an object reference or a number in an array. Be aware that the reference to this tag value is kept until the event is called and can cause memory reference cycles.

4.114.7 exportMT

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Starts the synchronous execution of an export session.
Notes: Same as exportAsynchronously, but run synchronous, so the plugin waits.

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.

4.114.8 exportSessionWithAsset(asset as AVAssetMBS) as AVAssetExportSessionMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Returns an asset export session configured with a specified asset and preset.
Notes: asset: The asset you want to export.

Returns an asset export session initialized to export asset.

4.114.9 metadata as AVMetadataItemMBS()

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Returns the metadata to be written to the output file by the export session.

4.114.10 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Sets the metadata to be written to the output file by the export session.
Notes: If the array is empty, any existing metadata in the exported asset will be translated as accurately as possible into the appropriate metadata key space for the output file and written to the output.

4.114.11 Properties

4.114.12 asset as AVAssetMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The asset with which the export session was initialized. (read-only)
Notes: (Read only property)

4.114.13 audioMix as AVAudioMixMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Indicates whether non-default audio mixing is enabled for export, and supplies the parameters for audio mixing.
Notes: (Read and Write property)

4.114.14 audioSettings as Dictionary

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The audio settings to use.
Example:

```
dim encoder as SDAVAssetExportSessionMBS // your exporter instance

Dim audioSettings As New Dictionary

audioSettings.Value(AVFoundationMBS.AVFormatIDKey) = OSTypeFromStringMBS(AVFoundationMBS.kAudioFormatMPEG4AAC)
audioSettings.Value(AVFoundationMBS.AVNumberOfChannelsKey) = 2
audioSettings.Value(AVFoundationMBS.AVSampleRateKey) = 44100
audioSettings.Value(AVFoundationMBS.AVEncoderBitRateKey) = 128000
encoder.audioSettings = audioSettings
```

Notes: Please create new dictionary, set values and assign the final dictionary to this property.
If you query the settings, you get a copy of the dictionary.
(Read and Write property)

4.114.15 error as NSErrorMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Describes the error that occurred if the export status is AVAssetExportSessionStatusFailed or AVAssetExportSessionStatusCancelled. (read-only)
Notes: If there is no error to report, the value of this property is nil.
(Read only property)

4.114.16 Handle as Integer

Plugin Version: 20.3, Platform: macOS, Targets: All.

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

4.114.17 outputFile as FolderItem

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The file of the export session's output.
Example:

```
dim encoder as SDAVAssetExportSessionMBS // your exporter instance

encoder.outputFile = SpecialFolder.Desktop.Child("output.mp4")
```

Notes: (Read and Write property)

4.114.18 outputFileType as String

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The type of file to be written by the session.
Example:

```
dim encoder as SDAVAssetExportSessionMBS // your exporter instance

encoder.outputFileType = AVFoundationMBS.AVFileTypeMPEG4
```

Notes: The value is a UTI string corresponding to the file type to use when writing the asset. For a list of constants specifying UTIs for standard file types, see AV Foundation Constants Reference.
(Read and Write property)

4.114.19 outputURL as String

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The URL of the export session's output.
Notes: (Read and Write property)

4.114.20 progress as Double

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The progress of the export on a scale from 0 to 1. (read-only)
Notes: A value of 0 means the export has not yet begun, 1 means the export is complete.
(Read only property)

4.114.21 shouldOptimizeForNetworkUse as Boolean

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Indicates whether the movie should be optimized for network use.
Notes: (Read and Write property)

4.114.22 Status as Integer

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The status of the export session. (read-only)
Notes: See status constants.
(Read only property)

4.114.23 timeRange as CMTimeRangeMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The time range to be exported from the source.
Notes: The default time range of an export session is kCMTimeZero to kCMTimePositiveInfinity, meaning that (modulo a possible limit on file length) the full duration of the asset will be exported.
(Read and Write property)

4.114.24 videoComposition as AVVideoCompositionMBS

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Indicates whether video composition is enabled for export, and supplies the instructions for video composition.
Notes: (Read and Write property)

4.114.25 videoInputSettings as Dictionary

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The video input settings to use.
Notes: Please create new dictionary, set values and assign the final dictionary to this property.
If you query the settings, you get a copy of the dictionary.
(Read and Write property)

4.114.26 videoSettings as Dictionary

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: The video settings to use.
Example:

```
dim encoder as SDAVAssetExportSessionMBS // your exporter instance

Dim CompressionProperties As New Dictionary
CompressionProperties.Value(AVFoundationMBS.AVVideoAverageBitRateKey) = 6000000
CompressionProperties.Value(AVFoundationMBS.AVVideoProfileLevelKey) = AVFoundationMBS.AVVideoProfileLevelH264High40

Dim videoSettings As New Dictionary

videoSettings.Value(AVFoundationMBS.AVVideoCodecKey) = AVFoundationMBS.AVVideoCodecH264
videoSettings.Value(AVFoundationMBS.AVVideoWidthKey) = 1920
videoSettings.Value(AVFoundationMBS.AVVideoHeightKey) = 1080
videoSettings.Value(AVFoundationMBS.AVVideoCompressionPropertiesKey) = CompressionProperties
encoder.videoSettings = videoSettings
```

Notes: Please create new dictionary, set values and assign the final dictionary to this property.
If you query the settings, you get a copy of the dictionary.
(Read and Write property)

4.114.27 Events

4.114.28 exportAsynchronouslyCompleted

Plugin Version: 20.3, Platform: macOS, Targets: .

Function: Returns the identifiers compatible with a given asset.
Notes: asset: An asset that is ready to be exported.

Returns an array of strings representing the identifiers compatible with asset.

The array is a complete list of the valid identifiers that can be used with exportSessionWithAsset with the specified asset.

Not all export presets are compatible with all assets. For example, a video-only asset is not compatible with an audio-only preset. This method returns only the identifiers for presets that are compatible with the given asset.

In order to ensure that the setup and running of an export operation will succeed using a given preset, you should not make significant changes to the asset (such as adding or deleting tracks) between retrieving compatible identifiers and performing the export operation.

If the asset's tracks are not currently loaded, they may be loaded by this method before any checks are performed.

4.114.29 Constants

Status Constants

Constant	Value	Description
StatusCancelled	5	Indicates that the export session was cancelled.
StatusCompleted	3	Indicates that the export session completed successfully.
StatusExporting	2	Indicates that the export session is in progress.
StatusFailed	4	Indicates that the export session failed.
StatusUnknown	0	Indicates that the status is unknown.
StatusWaiting	1	Indicates that the session is waiting to export more data.

4.115 class Sound

4.115.1 class Sound

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in sound class in Xojo.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp4")
dim s as sound = f.OpenAsSound
dim a as AVAudioPlayerMBS = s.AVAudioPlayerMBS
MsgBox str(a.duration)+" seconds long"
call a.play
```

4.115.2 Methods

4.115.3 AVAudioPlayerMBS as AVAudioPlayerMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The audio player for this sound.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.mp4")
dim s as sound = f.OpenAsSound
dim a as AVAudioPlayerMBS = s.AVAudioPlayerMBS
MsgBox str(a.duration)+" seconds long"
call a.play
```

Notes: Xojo uses AVAudioPlayer for playing sounds in version 2013r4 and newer.
This function returns the player object, so you change more settings.

Chapter 5

AVFoundationNode

5.1 class AVAudio3DPointMBS

5.1.1 class AVAudio3DPointMBS

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

Function: Class representing a point in 3D space.
Notes: This class is used by classes dealing with 3D audio such as those that adopt the AVAudioMixing protocol and the AVAudioEnvironmentNode class and represents a point in 3D space, in meters.

5.1.2 Methods

5.1.3 Constructor(x as Double = 0.0, y as Double = 0.0, z as Double = 0.0)

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

Function: The constructor.

5.1.4 Properties

5.1.5 x as Double

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

Function: The location on the x-axis, in meters.
Notes: (Read and Write property)

5.1.6 y as Double

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

Function: The location on the y-axis, in meters.
Notes: (Read and Write property)

5.1.7 z as Double

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

Function: The location on the z-axis, in meters.
Notes: (Read and Write property)

5.2 class AVAudioBufferMBS

5.2.1 class AVAudioBufferMBS

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

Function: The AVAudioBuffer class represents a buffer of audio data and its format.
Notes: This is 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 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.2.4 Constructor

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

Function: The private constructor.

5.2.5 copy as AVAudioBufferMBS

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

Function: Creates a read only copy of the buffer.

5.2.6 mutableCopy as AVAudioBufferMBS

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

Function: Creates a mutable copy of the buffer.

5.2.7 Properties

5.2.8 byteCapacity as UInt32

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The buffer's capacity in bytes.
Notes: Available on MacOS 10.13 or newer.
(Read and Write property)

5.2.9 byteLength as UInt32

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The current number of valid bytes in the buffer.
Notes: Can be changed as part of an operation that modifies the contents.

Available on MacOS 10.13 or newer.
(Read only property)

5.2.10 format as AVAudioFormatMBS

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

Function: The format of the audio in the buffer.
Notes: (Read only property)

5.2.11 Handle as Integer

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

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

5.3 class AVAudioChannelLayoutMBS

5.3.1 class AVAudioChannelLayoutMBS

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

Function: The AVAudioChannelLayout class describes the roles of a set of audio channels.
Example:

```
Function StereoLeftFormat() As AVAudioFormatMBS
// stereo format, left channel only

dim qlayout as new QTAudioChannelLayoutMBS

dim d1 as new QTAudioChannelDescriptionMBS
dim d2 as new QTAudioChannelDescriptionMBS

d1.ChannelFlags = QTAudioChannelDescriptionMBS.kAudioChannelFlags_AllOff
d1.ChannelLabel = QTAudioChannelDescriptionMBS.kAudioChannelLabel_Discrete_3

d2.ChannelFlags = QTAudioChannelDescriptionMBS.kAudioChannelFlags_AllOff
d2.ChannelLabel = QTAudioChannelDescriptionMBS.kAudioChannelLabel_Unused

qlayout.ChannelLayoutTag = QTAudioChannelLayoutMBS.kAudioChannelLayoutTag_UseChannelDescriptions

qlayout.NumberChannelDescriptions=2
qlayout.ChannelDescriptions(0)=d1
qlayout.ChannelDescriptions(1)=d2

dim layout as new AVAudioChannelLayoutMBS(qlayout)
dim pf as new AVAudioFormatMBS(44100, layout)

Return pf
End Function
```

Notes: This object is a thin wrapper for the AudioChannelLayout structure.

5.3.2 Methods

5.3.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.3.4 Constructor(Layout as QTAudioChannelLayoutMBS)

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

Function: Initialize an audio channel layout object using an existing one.
Example:

```
Function StereoFormat() As AVAudioFormatMBS
// stereo format with standard layout tag
dim qlayout as new QTAudioChannelLayoutMBS

qlayout.ChannelLayoutTag = QTAudioChannelLayoutMBS.kAudioChannelLayoutTag_Stereo

dim layout as new AVAudioChannelLayoutMBS(qlayout)
dim pf as new AVAudioFormatMBS(44100, layout)

Return pf
End Function
```

Notes: If the provided audio channel layout object‚Äôs tag is kAudioChannelLayoutTag_UseChannelDescriptions, this initializer attempts to convert it to a more specific tag.
See also:

- 5.3.5 Constructor(LayoutTag as Integer) 760

5.3.5 Constructor(LayoutTag as Integer)

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

Function: Initialize an audio channel layout object using a layout tag.
Example:

```
Function StereoFormat() As AVAudioFormatMBS
// stereo format with standard layout tag

dim layout as new AVAudioChannelLayoutMBS(QTAudioChannelLayoutMBS.kAudioChannelLayoutTag_Stereo)
dim pf as new AVAudioFormatMBS(44100, layout)

Return pf
End Function
```

Notes: See QTAudioChannelLayoutMBS class for constants.
See also:

- 5.3.4 Constructor(Layout as QTAudioChannelLayoutMBS) 760

5.3.6 isEqual(other as AVAudioChannelLayoutMBS) as boolean

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

Function: Determines whether another audio channel layout is exactly equal to this layout.
Notes: other: The AVAudioChannelLayout object to compare against.

Returns true if they are equal; otherwise false.

5.3.7 layoutWithLayout(Layout as QTAudioChannelLayoutMBS) as AVAudioChannelLayoutMBS

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

Function: Creates a new audio channel layout object from an existing one.

5.3.8 layoutWithLayoutTag(LayoutTag as Integer) as AVAudioChannelLayoutMBS

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

Function: Creates a new audio channel layout object using an audio channel layout tag.
Notes: If the provided audio channel layout object,Äôs tag is kAudioChannelLayoutTag_UseChannelDescriptions, this initializer attempts to convert it to a more specific tag.

5.3.9 Properties

5.3.10 channelCount as Integer

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

Function: The number of channels of audio data.
Notes: (Read only property)

5.3.11 Handle as Integer

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

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

5.3.12 layout as QTAudioChannelLayoutMBS

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

Function: The underlying audio channel layout.
Notes: See QTAudioChannelLayoutMBS class for details.
(Read only property)

5.3.13 layoutTag as Integer

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

Function: The audio channel‚Äôs underlying layout tag.
Notes: See QTAudioChannelLayoutMBS class for constants.
(Read only property)

5.4 class AVAudioComponentDescriptionMBS

5.4.1 class AVAudioComponentDescriptionMBS

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

Function: Identifying information for an audio component.

5.4.2 Methods

5.4.3 Constructor(componentType as string = "", componentSubType as string = "", componentManufacturer as string = "", componentFlags as UInt32 = 0, componentFlagsMask as UInt32 = 0)

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

Function: The constructor.

5.4.4 Properties

5.4.5 componentFlags as UInt32

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

Function: The flags.
Notes: Set this value to zero.
(Read and Write property)

5.4.6 componentFlagsMask as UInt32

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

Function: The mask of flags set.
Notes: Set this value to zero.
(Read and Write property)

5.4.7 componentManufacturer as String

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

Function: The unique vendor identifier, registered with Apple, for the audio component.
Notes: (Read and Write property)

5.4.8 componentSubType as String

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

Function: A 4-byte code that you can use to indicate the purpose of a component.
Notes: For example, you could use lpas or lowp as a mnemonic indication that an audio unit is a low-pass filter.
(Read and Write property)

5.4.9 componentType as String

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

Function: A unique 4-byte code identifying the interface for the component.
Notes: (Read and Write property)

5.5 class AVAudioEngineMBS

5.5.1 class AVAudioEngineMBS

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

Function: The AVAudioEngine class defines a group of connected AVAudioNode objects, known as audio nodes.
Notes: You use audio nodes to generate audio signals, process them, and perform audio input and output.

You create each audio node separately and attach it to the audio engine. You can perform all audio node operations during runtime—connecting them, disconnecting them, and removing them—with only minor limitations:

- Reconnect audio nodes only when they are upstream of a mixer..
- If you remove an audio node that has differing input and output channel counts, or which is a mixer, the result will likely be a broken graph.

Blog Entries

- MBS Xojo Plugins Version 21.0 News
- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0
- MBS Xojo Plugins, version 21.0pr6
- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.3

Xojo Developer Magazine

- 19.2, page 10: News

5.5.2 Methods

5.5.3 attachedNodes as AVAudioNodeMBS()

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Set of all nodes attached to the engine.
Notes: Available in macOS 10.15 or newer.
Available in iOS 10.13 or newer.

5.5.4 attachNode(node as AVAudioNodeMBS)

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

Function: Attaches a new audio node to the audio engine.
Notes: node: The audio node to be attached to the audio engine.

To support the instantiation of arbitrary AVAudioNode subclasses, instances are created externally to the audio engine but are unusable until attached to the audio engine using this method.

5.5.5 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.5.6 AVAudioEngineConfigurationChangeNotification as String

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

Function: Posted when the audio engine configuration changes.
Notes: When the audio engine,Äôs I/O unit observes a change to the audio input or output hardware,Äôs channel count or sample rate, the audio engine stops, uninitializes itself, and issues this notification.

5.5.7 connect(node1 as AVAudioNodeMBS, node2 as AVAudioNodeMBS, bus1 as Integer, bus2 as Integer, format as AVAudioFormatMBS)

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

Function: Establish a connection between two audio nodes, specifying the input and output busses.
Notes: node1: The source audio node.
node2: The destination audio node.
bus1: The output bus of the source audio node.
bus2: The input bus of the destination audio node.
format: If not nil, the format of the source audio node,Äôs output bus is set to this AVAudioFormat object. In all cases, the format of the destination audio node,Äôs input bus is set to match that of the source audio node,Äôs output bus.

Audio nodes have input and output busses (AVAudioNodeBus). Use this method to establish connections between audio nodes. Connections are always one-to-one, never one-to-many or many-to-one.
See also:

- 5.5.8 connect(node1 as AVAudioNodeMBS, node2 as AVAudioNodeMBS, format as AVAudioFormatMBS) 767

5.5.8 connect(node1 as AVAudioNodeMBS, node2 as AVAudioNodeMBS, format as AVAudioFormatMBS)

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

Function: Establishes a connection between two audio nodes.
Notes: node1: The source audio node.
node2: The destination audio node.

format: If not nil, the format of the source audio node,Äôs output bus is set to this AVAudioFormat object. In all cases, the format of the destination audio node,Äôs input bus is set to match that of the source audio node,Äôs output bus.

This method calls connect method using bus 0 for the source audio node, and bus 0 for the destination audio node, except in the case of a destination which is a mixer, in which case the destination is the mixer,Äôs nextAvailableInputBus.
See also:

- 5.5.7 connect(node1 as AVAudioNodeMBS, node2 as AVAudioNodeMBS, bus1 as Integer, bus2 as Integer, format as AVAudioFormatMBS) 766

5.5.9 Constructor

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

Function: The constructor.

5.5.10 Destructor

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

Function: The destructor.

5.5.11 detachNode(node as AVAudioNodeMBS)

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

Function: Detaches a audio node previously attached to the audio engine.
Notes: node: The audio node to be detached from the audio engine.

If necessary, the audio engine will safely disconnect the audio node before detaching it.

5.5.12 disableManualRenderingMode

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Sets the engine to render to or from an audio device.

5.5.13 disconnectNodeInput(node as AVAudioNodeMBS)

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

Function: Removes all input connections of the audio node.
Notes: node: The audio node whose inputs you want to disconnect.

Connections are broken on each of the audio node,Äôs input buses.
See also:

- 5.5.14 disconnectNodeInput(node as AVAudioNodeMBS, bus as Integer) 768

5.5.14 disconnectNodeInput(node as AVAudioNodeMBS, bus as Integer)

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

Function: Removes the input connection of an audio node on the specified bus.
Notes: node: The audio node whose input is to be disconnected.
bus: The destination,Äôs input bus to be disconnected.
See also:

- 5.5.13 disconnectNodeInput(node as AVAudioNodeMBS) 768

5.5.15 disconnectNodeOutput(node as AVAudioNodeMBS)

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

Function: Remove all output connections of an audio node.
Notes: node: The audio node whose outputs are to be disconnected.
See also:

- 5.5.16 disconnectNodeOutput(node as AVAudioNodeMBS, bus as Integer) 769

5.5.16 disconnectNodeOutput(node as AVAudioNodeMBS, bus as Integer)

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

Function: Remove the output connection of an audio node on the specified bus.
Notes: node: The audio node whose output is to be disconnected.
bus: The destination,Äôs output bus to disconnect.
See also:

- 5.5.15 disconnectNodeOutput(node as AVAudioNodeMBS) 768

5.5.17 enableManualRenderingMode(mode as Integer, pcmFormat as AVAudioFormatMBS, maximumFrameCount as UInt32, byref error as NSErrorMBS) as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Sets the engine to operate in manual rendering mode with the specified render format and maximum frame count.
Notes: Use this method to configure the engine to render in response to requests from the client. You must stop the engine before calling this method. The render format must be a PCM format and match the format of the rendering buffer.
The input data in manual rendering mode can be supplied through the source nodes. See AVAudioPlayerNode and AVAudioInputNode.

5.5.18 pause

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

Function: Pauses the audio engine.
Notes: Stops the flow of audio through the audio engine but does not deallocate the resources allocated by prepare.
You resume the audio engine by invoking startAndReturnError again.

5.5.19 prepare

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

Function: Prepares the audio engine for starting.
Notes: This method preallocates many of the resources that the audio engine requires in order to start.

Use it in order to start audio input or output more responsively.

5.5.20 renderOffline(numberOfFrames as UInt32, buffer as AVAudioPCMBufferMBS, byref error as NSErrorMBS) as Integer

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: Makes a render call to the engine operating in the offline manual rendering mode.

5.5.21 reset

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

Function: Resets all of the audio nodes in the audio engine.
Notes: This methods resets all of the audio nodes in the audio engine. Use it, for example, to silence reverb and delay tails.

5.5.22 startAndReturnError(byref error as NSErrorMBS) as Boolean

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

Function: Starts the audio engine.
Notes: Error: Returns, by-reference, a description of the error.

Returns true if the audio engine started successfully; otherwise, false.
This method invokes the prepare method if it has not already been called since stop was invoked.
It then starts the audio hardware via the AVAudioInputNode and/or AVAudioOutputNode instances in the audio engine.

Errors can occur when:

- There is problem in the structure of the graph. Input can't be routed to output or to a recording tap through converter type nodes.
- An AVAudioSession error occurred
- The driver failed to start the hardware

5.5.23 stop

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

Function: Stop the audio engine.
Notes: Releases the resources allocated by the prepare method.

5.5.24 Properties

5.5.25 autoShutdownEnabled as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: When auto shutdown is enabled, the engine can start and stop the audio hardware dynamically, to conserve power. This is the enforced behavior on watchOS and can be optionally enabled on other platforms.
Notes: To conserve power, it is advised that the client pause/stop the engine when not in use.
But when auto shutdown is enabled, the engine will stop the audio hardware if it was running idle for a certain duration, and restart it later when required.
Note that, because this operation is dynamic, it may affect the start times of the source nodes (e.g. AVAudioPlayerNodeMBS), if the engine has to resume from its shutdown state.

On watchOS, auto shutdown is always enabled. On other platforms, it is disabled by default, but the client can enable it if needed.

This property is applicable only when the engine is rendering to/from an audio device. If the value is changed when the engine is in manual rendering mode, it will take effect whenever the engine is switched to render to/from the audio device.

Available on MacOS 10.13 or newer.
(Read and Write property)

5.5.26 Handle as Integer

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

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

5.5.27 InManualRenderingMode as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Whether or not the engine is operating in manual rendering mode, i.e. not connected to an audio device and rendering in response to the requests from the client.

Notes: Manual rendering is currently not supported in MBS Plugins.

Available on MacOS 10.13 or newer.
(Read only property)

5.5.28 inputNode as AVAudioInputNodeMBS

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

Function: The audio engine‚Äôs singleton input audio node.
Notes: The audio engine creates a singleton on demand when inputNode is first accessed. To receive input, connect another audio node from the output of the input audio node, or create a recording tap on it.

The AVAudioSesssion instance and/or the availability of hardware features determine whether an app can perform input. Check the input format of input node (specifically, the hardware format) for a non-zero sample rate and channel count to see if input is enabled.
(Read only property)

5.5.29 mainMixerNode as AVAudioMixerNodeMBS

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

Function: The audio engine‚Äôs optional singleton main mixer node.
Notes: When the property is first accessed the audio engine constructs a singleton main mixer and connects it to the outputNode on demand. You can then connect additional audio nodes to the mixer.

By default, the mixer's output format (sample rate and channel count) will track the format of the output node. It is possible to make the connection explicitly with a different format.
(Read only property)

5.5.30 manualRenderingFormat as AVAudioFormatMBS

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: The render format of the engine in manual rendering mode.
Notes: (Read only property)

5.5.31 manualRenderingMaximumFrameCount as UInt32

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: The maximum number of PCM sample frames the engine can produce in any single render call in manual rendering mode.
Notes: If you get this property when the engine isn't in manual rendering mode, it returns zero.
(Read only property)

5.5.32 manualRenderingMode as Integer

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: The manual rendering mode configured on the engine.
Notes: Either ManualRenderingModeOffline or ManualRenderingModeRealtime.
(Read only property)

5.5.33 manualRenderingSampleTime as Int64

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: An indication of where the engine is on its render timeline in manual rendering mode.
Notes: (Read only property)

5.5.34 outputNode as AVAudioOutputNodeMBS

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

Function: The audio engine,Äôs singleton output audio node.
Notes: When this property is first accessed the audio engine creates a singleton on demand. Connect another audio node to the input of the output audio node, or obtain a mixer that is connected by default, using the mainMixerNode property.

The AVAudioSesssion instance and/or the availability of hardware features determine whether an app can perform output. Check the output format of output node (specifically, the hardware format) for a non-zero sample rate and channel count to see if output is enabled.
(Read only property)

5.5.35 running as Boolean

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

Function: Returns the audio engine,Äôs running state.
Notes: The value is true if the audio engine is running, otherwise, false.
(Read only property)

5.5.36 Events

5.5.37 ConfigurationChanged(notification as NSNotificationMBS)

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

Function: Posted when the audio engine configuration changes.
Notes: When the audio engine,Äôs I/O unit observes a change to the audio input or output hardware,Äôs channel count or sample rate, the audio engine stops, uninitializes itself, and issues this notification.

5.5.38 Constants

Error Codes

Constant	Value	Description
ManualRenderingErrorInitialized	-80801	An operation that can,Äôt be performed because the engine hasn,Äôt stop running.
ManualRenderingErrorInvalidMode	-80800	An operation that can,Äôt be performed because the engine is not in man rendering mode or the right variant of it.
ManualRenderingErrorNotRunning	-80802	An operation that can,Äôt be performed because the engine hasn,Äôt star running.

Rendering Modes

Constant	Value	Description
ManualRenderingModeOffline	0	An engine that operates in an offline mode.
ManualRenderingModeRealtime	1	An engine that operates under real-time constraints and doesn't make blocking calls while rendering.

Manual Rendering Status

Constant	Value	Description
ManualRenderingStatusCannotDoInCurrentContext	2	An operation that can't be performed under current co try again later.
ManualRenderingStatusError	-1	A problem that occurred during rendering, resulting in
ManualRenderingStatusInsufficientDataFromInputNode	1	A condition in which not enough input data was retur to satisfy the render request at the time of the request
ManualRenderingStatusSuccess	0	A status that indicates that all of the requested data wa

5.6 class AVAudioEnvironmentDistanceAttenuationParametersMBS

5.6.1 class AVAudioEnvironmentDistanceAttenuationParametersMBS

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

Function: The AVAudioEnvironmentDistanceAttenuationParameters class specifies the attenuation distance, the gradual loss in audio intensity, and characteristics.
Notes: 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 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.6.4 Constructor

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

Function: The private constructor.

5.6.5 Properties

5.6.6 distanceAttenuationModel as Integer

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

Function: The distance attenuation model describing the drop-off in gain as the source moves away from the listener.
Notes: The default value is the AVAudioEnvironmentDistanceAttenuationModelInverse attenuation model.
(Read and Write property)

5.6.7 Handle as Integer

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

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

5.6.8 maximumDistance as Double

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

Function: The distance, in meters, beyond which no further attenuation is applied.
Notes: The default value is 100000.0 meters.
This property is relevant for the AVAudioEnvironmentDistanceAttenuationModelInverse attenuation model.
(Read and Write property)

5.6.9 referenceDistance as Double

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

Function: The minimum distance, in meters, at which attenuation is applied.
Notes: The default value is 1.0 meter.

This property is relevant for the AVAudioEnvironmentDistanceAttenuationModelInverse and AVAudioEnvironmentDistanceAttenuationModelLinear attenuation models.
(Read and Write property)

5.6.10 rolloffFactor as Double

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

Function: Determines the attenuation curve.
Notes: A higher value results in a steeper attenuation curve. The default value is 1.0. The value must be greater than 0.0.

This property is relevant for the AVAudioEnvironmentDistanceAttenuationModelExponential, AVAudioEnvironmentDistanceAttenuationModelInverse, and AVAudioEnvironmentDistanceAttenuationModelLinear attenuation models.
(Read and Write property)

5.6.11 Constants

Distance attenuation models

Constant	Value	Description
DistanceAttenuationModelExponential	1	Calculated as distanceGain = (distance / referenceDistance) ^(-rolloffFa
DistanceAttenuationModelInverse	2	Calculated as distanceGain = referenceDistance / (referenceDistance + r Factor * (distance –referenceDistance)).
DistanceAttenuationModelLinear	3	Calculated as distanceGain = (1 –rolloffFactor * (distance –referenceDis / (maximumDistance –referenceDistance)).

5.7 class AVAudioEnvironmentNodeMBS

5.7.1 class AVAudioEnvironmentNodeMBS

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

Function: The AVAudioEnvironmentNode class is a mixer node that simulates a 3D audio environment.
Notes: Any node that conforms to the AVAudioMixing protocol (for example, AVAudioPlayerNode) can act as a source in this environment.

The environment has an implicit ,Äúlistener,Äù. By controlling the listener,Äôs position and orientation, the application controls the way the user experiences the virtual world. In addition, this node also defines properties for distance attenuation and reverberation that help characterize the environment.

It is important to note that only inputs with a mono channel connection format to the environment node are spatialized. If the input is stereo, the audio is passed through without being spatialized. Currently inputs with connection formats of more than 2 channels are not supported.

In order to set the environment node,Äôs output to a multichannel format, use an AVAudioFormatMBS having one of the following Audio Channel Layout Tags.

- kAudioChannelLayoutTag_AudioUnit_4
- kAudioChannelLayoutTag_AudioUnit_5_0
- kAudioChannelLayoutTag_AudioUnit_6_0
- kAudioChannelLayoutTag_AudioUnit_7_0
- kAudioChannelLayoutTag_AudioUnit_7_0_Front
- kAudioChannelLayoutTag_AudioUnit_8

Subclass of the AVAudioNodeMBS class.

5.7.2 Methods

5.7.3 Constructor

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

Function: The constructor.

5.7.4 Properties

5.7.5 distanceAttenuationParameters as AVAudioEnvironmentDistanceAttenuationParametersMBS

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

Function: The distance attenuation parameters for the environment.
Notes: (Read only property)

5.7.6 nextAvailableInputBus as Integer

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

Function: Finds an unused input bus.
Notes: This method finds and returns the first input bus to which no other node is connected.
(Read only property)

5.7.7 obstruction as Double

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

Function: Simulates filtering of the direct path of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Only the direct path of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.8 occlusion as Double

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

Function: Simulates filtering of the direct and reverb paths of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Both the direct and reverb paths of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.9 outputVolume as Double

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

Function: The mixer's output volume.
Notes: This accesses the mixer's output volume (0.0-1.0, inclusive).
(Read and Write property)

5.7.10 pan as Double

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

Function: The buss's stereo pan.
Notes: The default value is 0.0. A value in the range -1.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.11 position as AVAudio3DPointMBS

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

Function: The location of the source in the 3D environment.
Notes: The coordinates are specified in meters. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.12 rate as Double

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

Function: Changes the playback rate of the input signal.
Notes: A value of 2.0 results in the output audio playing one octave higher. A value of 0.5, results in the output audio playing one octave lower.

The default value is 1.0. The range of valid values is 0.5 to 2.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.13 renderingAlgorithm as Integer

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

Function: Type of rendering algorithm used.
Notes: Depending on the current output format of the AVAudioEnvironmentNode instance, only a subset of the rendering algorithms may be supported. An array of valid rendering algorithms is retrieved by calling the applicableRenderingAlgorithms function of the AVAudioEnvironmentNode instance.

The default rendering algorithm is AVAudio3DMixingRenderingAlgorithmEqualPowerPanning. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.14 reverbBlend as Double

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

Function: Controls the blend of dry and reverb processed audio.
Notes: This property controls the amount of the source,Äôs audio that will be processed by the reverb by the AVAudioEnvironmentNode instance. A value of 0.5 will result in an equal blend of dry and processed (wet) audio.

The default is 0.0. The range of valid values is 0.0 (completely dry) to 1.0 (completely wet). This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.7.15 reverbParameters as AVAudioEnvironmentReverbParametersMBS

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

Function: The reverb parameters for the environment.
Notes: (Read only property)

5.7.16 volume as Double

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

Function: The buss's input volume.
Notes: The default value is 1.0. The range of valid values are 0.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode and AVAudioMixerNode class mixers.
(Read and Write property)

5.8 class AVAudioEnvironmentReverbParametersMBS

5.8.1 class AVAudioEnvironmentReverbParametersMBS

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

Function: The AVAudioEnvironmentReverbParameters class encapsulates the parameters that you use to control the reverb of the AVAudioEnvironmentNode class.
Notes: Reverberation can be used to simulate the acoustic characteristics of an environment. The AVAudioEnvironmentNode class has a built-in reverb that describes the space that the listener is in.

The reverb also has a single filter that sits at the end of the chain. You can use this filter to shape the overall sound of the reverb. For instance, select one of the reverb presets to simulate the general space and then use the filter to brighten or darken the overall sound.

You cannot create a standalone instance of AVAudioEnvironmentReverbParameters. Only an instance vended by a source object, such as an AVAudioEnvironmentNode instance is valid.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.8.2 Methods

5.8.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.8.4 Constructor

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

Function: The private constructor.

5.8.5 loadFactoryReverbPreset(preset as Integer)

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

Function: Loads one of the reverbs factory presets.
Notes: Loading a factory reverb preset changes the sound of the reverb. This is independent of the filter

which follows the reverb in the signal chain.

5.8.6 Properties

5.8.7 enable as Boolean

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

Function: Enables the reverberation.
Notes: The default value is false.
(Read and Write property)

5.8.8 filterParameters as AVAudioUnitEQFilterParametersMBS

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

Function: The filter that applies to the output of the reverb.
Notes: (Read only property)

5.8.9 Handle as Integer

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

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

5.8.10 level as Double

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

Function: The master level, in decibels, of the reverb.
Notes: The default value is 0.0. The values must be within the range of -40 to 40 dB.
(Read and Write property)

5.9 class AVAudioFileMBS

5.9.1 class AVAudioFileMBS

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

Function: The AVAudioFile class represents an audio file that can be opened for reading or writing.
Notes: Regardless of the file‚Äôs actual format, you read and write the file using AVAudioPCMBuffer objects, that contain samples in AVAudioCommonFormat, referred to as the file‚Äôs processing format. Conversions are performed to and from the file‚Äôs actual format.

Reads and writes are always sequential, but random access is possible by setting the framePosition property.
Blog Entries

- Write audio file with samples using AVFoundation
- MBS Xojo / Real Studio Plugins, version 15.4pr1

5.9.2 Methods

5.9.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.9.4 Constructor(File as FolderItem, byref error as NSErrorMBS)

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

Function: Opens a file for reading.
Notes: file: The path of the file to read.
Error: Returns, by-reference, a description of the error, if an error occurs.
See also:

- 5.9.5 Constructor(File as FolderItem, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 785
- 5.9.6 Constructor(File as FolderItem, settings as Dictionary, byref error as NSErrorMBS) 785
- 5.9.7 Constructor(File as FolderItem, settings as Dictionary, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 786

5.9.5 Constructor(File as FolderItem, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS)

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

Function: Open a file for reading using a specified processing format.
Notes: file: The path of the file to read.
format: The processing format to use when reading from the file.
interleaved: Whether to use an interleaved processing format.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns an initialized audio file object for reading.

The file format me be is deinterleaved float, that is AVAudioPCMFormatFloat32. The processing format refers to the buffers read from the file. The content is read and converted from the file format to the processing format. The processing format must be at the same sample rate as the actual file contents and must be linear PCM, whether or not the processing buffer is interleaved float is determined by the interleaved parameter.
See also:

- 5.9.4 Constructor(File as FolderItem, byref error as NSErrorMBS) 784
- 5.9.6 Constructor(File as FolderItem, settings as Dictionary, byref error as NSErrorMBS) 785
- 5.9.7 Constructor(File as FolderItem, settings as Dictionary, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 786

5.9.6 Constructor(File as FolderItem, settings as Dictionary, byref error as NSErrorMBS)

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

Function: Open a file for writing.
Notes: file: The path of the file to create for writing.
settings: The format of the file to create. (See the settings property in the AVAudioRecorder class.)
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns an initialized audio file object for writing.

The file type to create is inferred from the file extension of fileURL. This method will overwrite a file at the specified URL if a file already exists.

The file is opened for writing using the standard format, AVAudioPCMFormatFloat32.
See also:

- 5.9.4 Constructor(File as FolderItem, byref error as NSErrorMBS) 784
- 5.9.5 Constructor(File as FolderItem, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 785
- 5.9.7 Constructor(File as FolderItem, settings as Dictionary, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 786

5.9.7 Constructor(File as FolderItem, settings as Dictionary, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS)

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

Function: Open a file for writing using a specified processing format.
Notes: file: The path to write the file.
settings: The format of the file to create.
format: The processing format to use when writing to the file.
interleaved: Whether to use an interleaved processing format.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns an initialized audio file object for writing.

The file type to create is inferred from the file extension of file.
This method will overwrite a file at the specified URL if a file already exists.
See also:

- 5.9.4 Constructor(File as FolderItem, byref error as NSErrorMBS) 784
- 5.9.5 Constructor(File as FolderItem, commonFormat as Integer, Interleaved as Boolean, byref error as NSErrorMBS) 785
- 5.9.6 Constructor(File as FolderItem, settings as Dictionary, byref error as NSErrorMBS) 785

5.9.8 fileDuration(file as folderItem) as Double

Plugin Version: 15.4, Platform: macOS, Targets: All.

Function: Queries duration of an audio file.
Notes: This uses different AVAudioFile API than rest of AVAudioFileMBS class.
We discovered a bug in Apples frameworks to report wrong length for some MP3 files and this function seems to be closer to the real duration of the file.

5.9.9 readIntoBuffer(buffer as AVAudioPCMBufferMBS, byref error as NSErrorMBS) as Boolean

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

Function: The buffer from which to read into the file.
Notes: buffer: The buffer from which to read the file. Its format must match the file,Äôs processing format.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true, if the read was successful; otherwise false.

Reading sequentially from the framePosition property, attempts to fill the buffer to its capacity. On return, the buffer,Äôs length property indicates the number of sample frames successfully read.
See also:

- 5.9.10 readIntoBuffer(buffer as AVAudioPCMBufferMBS, frameCount as Integer, byref error as NSErrorMBS) as Boolean 787

5.9.10 readIntoBuffer(buffer as AVAudioPCMBufferMBS, frameCount as Integer, byref error as NSErrorMBS) as Boolean

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

Function: Read a portion of a buffer.
Notes: buffer: The buffer from which to read the file. Its format must match the file,Äôs processing format.
frames: The number of frames to read.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true, if the read was successful; otherwise false.

Like the read method, but can be used to read fewer frames than the buffer frameCapacity.
See also:

- 5.9.9 readIntoBuffer(buffer as AVAudioPCMBufferMBS, byref error as NSErrorMBS) as Boolean 787

5.9.11 writeFromBuffer(buffer as AVAudioPCMBufferMBS, byref error as NSErrorMBS) as Boolean

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

Function: Write a buffer.
Notes: buffer: The buffer from which to write to the file. Its format must match the file,Äôs processing format.

Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true, if the read was successful.

Writes sequentially. The buffer's frameLength signifies how much of the buffer is to be written.

5.9.12 Properties

5.9.13 fileFormat as AVAudioFormatMBS

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

Function: The on-disk format of the file.
Notes: (Read only property)

5.9.14 FramePosition as Int64

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

Function: The position in the file at which the next read or write operation will occur.
Notes: Set the framePosition property to perform a seek before a read or write. A read or write operation advances the frame position value by the number of frames read or written.
(Read and Write property)

5.9.15 Handle as Integer

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

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

5.9.16 Length as Int64

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

Function: The number of sample frames in the file.
Notes: This can be computationally expensive to compute for the first time.
(Read only property)

5.9.17 processingFormat as AVAudioFormatMBS

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

Function: The processing format of the file.
Notes: (Read only property)

5.9.18 URL as String

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

Function: The URL the file is reading or writing.
Notes: (Read only property)

5.10 class AVAudioFormatMBS

5.10.1 class AVAudioFormatMBS

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

Function: The AVAudioFormat class wraps a Core Audio AudioStreamBasicDescription struct, with convenience initializers and accessors for common formats, including Core Audio,Äôs standard deinterleaved 32-bit floating point format.
Notes: Instances of this class are immutable.
Blog Entries

- Write audio file with samples using AVFoundation

5.10.2 Methods

5.10.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.10.4 Constructor(format as Integer, sampleRate as Double, channels as Integer, interleaved as Boolean)

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

Function: Initializes a newly allocated audio format instance.
Notes: format: The audio format. See AVAudioCommonFormat for values.
sampleRate: The sample rate, in hertz.
channels: The channel count.
interleaved: True if the audio format is interleaved; otherwise false.
See also:

- 5.10.5 Constructor(format as Integer, sampleRate as Double, interleaved as Boolean, layout as AVAudioChannelLayoutMBS) 791
- 5.10.6 Constructor(sampleRate as Double, channels as Integer) 791
- 5.10.7 Constructor(sampleRate as Double, layout as AVAudioChannelLayoutMBS) 791
- 5.10.8 Constructor(Settings as Dictionary) 792

5.10.5 Constructor(format as Integer, sampleRate as Double, interleaved as Boolean, layout as AVAudioChannelLayoutMBS)

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

Function: Initializes a newly allocated audio format instance with the specified auto format, sample rate, interleaved format and channel layout.
Notes: format: The audio format. See AVAudioCommonFormat for values.
sampleRate: The sample rate, in hertz.
interleaved: True if the audio format is interleaved; otherwise false.
layout: The channel layout. Must not be nil.
See also:

- 5.10.4 Constructor(format as Integer, sampleRate as Double, channels as Integer, interleaved as Boolean) 790
- 5.10.6 Constructor(sampleRate as Double, channels as Integer) 791
- 5.10.7 Constructor(sampleRate as Double, layout as AVAudioChannelLayoutMBS) 791
- 5.10.8 Constructor(Settings as Dictionary) 792

5.10.6 Constructor(sampleRate as Double, channels as Integer)

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

Function: Initializes a newly allocated audio format instance with the specified sample rate and channel count.
Notes: sampleRate: The sample rate, in hertz.
channels: The channel count.

The returned AVAudioFormat instance uses the AVAudioPCMFormatFloat32 format.
See also:

- 5.10.4 Constructor(format as Integer, sampleRate as Double, channels as Integer, interleaved as Boolean) 790
- 5.10.5 Constructor(format as Integer, sampleRate as Double, interleaved as Boolean, layout as AVAudioChannelLayoutMBS) 791
- 5.10.7 Constructor(sampleRate as Double, layout as AVAudioChannelLayoutMBS) 791
- 5.10.8 Constructor(Settings as Dictionary) 792

5.10.7 Constructor(sampleRate as Double, layout as AVAudioChannelLayoutMBS)

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

Function: Initializes a newly allocated audio format instance as a deinterleaved float with the specified sample rate and channel layout.
Example:

```
Function StereoFormat() As AVAudioFormatMBS
// stereo format with standard layout tag
dim qlayout as new QTAudioChannelLayoutMBS

qlayout.ChannelLayoutTag = QTAudioChannelLayoutMBS.kAudioChannelLayoutTag_Stereo

dim layout as new AVAudioChannelLayoutMBS(qlayout)
dim pf as new AVAudioFormatMBS(44100, layout)

Return pf
End Function
```

Notes: sampleRate: The sample rate, in hertz.
layout: The channel layout. Must not be nil.

The returned AVAudioFormat instance uses the AVAudioPCMFormatFloat32 format.
See also:

- 5.10.4 Constructor(format as Integer, sampleRate as Double, channels as Integer, interleaved as Boolean) 790
- 5.10.5 Constructor(format as Integer, sampleRate as Double, interleaved as Boolean, layout as AVAudioChannelLayoutMBS) 791
- 5.10.6 Constructor(sampleRate as Double, channels as Integer) 791
- 5.10.8 Constructor(Settings as Dictionary) 792

5.10.8 Constructor(Settings as Dictionary)

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

Function: Initializes a newly allocated audio format instance using a settings dictionary.
Notes: settings: The settings dictionary. See AV Foundation Audio Settings Constants for supported key/value pairs.
Many settings dictionary elements pertain to encoder settings, not the basic format, and will be ignored.
See also:

- 5.10.4 Constructor(format as Integer, sampleRate as Double, channels as Integer, interleaved as Boolean) 790
- 5.10.5 Constructor(format as Integer, sampleRate as Double, interleaved as Boolean, layout as AVAudioChannelLayoutMBS) 791

- 5.10.6 Constructor(sampleRate as Double, channels as Integer) 791
- 5.10.7 Constructor(sampleRate as Double, layout as AVAudioChannelLayoutMBS) 791

5.10.9 isEqual(other as AVAudioFormatMBS) as boolean

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

Function: Returns a Boolean value that indicates whether the audo format instance and a given object have identical format settings.
Notes: object: The object to be compared.

Returns true if the receiver and object are equal, otherwise false.

This method defines what it means for instances to be equal. The two objects are considered equal if and only if they return identical values for all the settings.

5.10.10 Properties

5.10.11 channelCount as Integer

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

Function: The number of channels of audio data.
Notes: (Read only property)

5.10.12 channelLayout as AVAudioChannelLayoutMBS

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

Function: The underlying audio channel layout.
Notes: The value is the underlying AVAudioChannelLayout instance.
(Read only property)

5.10.13 commonFormat as Integer

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

Function: Returns the common format identifier.
Notes: (Read only property)

5.10.14 Handle as Integer

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

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

5.10.15 Interleaved as Boolean

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

Function: Describes whether the samples are interleaved.
Notes: For non-PCM formats, this value is undefined.
(Read only property)

5.10.16 sampleRate as Double

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

Function: The audio format sampling rate in hertz.
Notes: (Read only property)

5.10.17 settings as Dictionary

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

Function: The format represented as a dictionary with keys from AVAudioSettings.
Notes: Not all formats representable by an AudioStreamBasicDescription struct (the underlying implementation) can be represented in a settings dictionary; in that case, nil is returned.
(Read only property)

5.10.18 Standard as Boolean

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

Function: Describes whether the format is deinterleaved native-endian float.
Notes: True if the format is deinterleaved native-endian float (AVAudioPCMFormatFloat32), otherwise false.
(Read only property)

5.10.19 Constants

Common audio formats

Constant	Value	Description
AVAudioOtherFormat	0	A format which is not 16/32bit integer or 32/64bit float.
AVAudioPCMFormatFloat32	1	Native-endian floats. This is the standard format.
AVAudioPCMFormatFloat64	2	Native-endian doubles.
AVAudioPCMFormatInt16	3	Signed 16-bit native-endian integers.
AVAudioPCMFormatInt32	4	Signed 32-bit native-endian integers.

5.11 class AVAudioInputNodeMBS

5.11.1 class AVAudioInputNodeMBS

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

Function: The AVAudioInputNode class represents a node that connects to the system's audio input.
Notes: This node has one element. The format of the input scope reflects the audio hardware sample rate and channel count. The format of the output scope is initially the same as that of the input, but you may set it to a different format, in which case the node will convert.
Subclass of the AVAudioIONodeMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.11.2 Methods

5.11.3 Constructor

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

Function: The private constructor.

5.11.4 Properties

5.11.5 obstruction as Double

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

Function: Simulates filtering of the direct path of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Only the direct path of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.6 occlusion as Double

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

Function: Simulates filtering of the direct and reverb paths of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Both the direct and reverb paths of sound between the

source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.7 pan as Double

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

Function: The buss's stereo pan.
Notes: The default value is 0.0. A value in the range -1.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.8 position as AVAudio3DPointMBS

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

Function: The location of the source in the 3D environment.
Notes: The coordinates are specified in meters. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.9 rate as Double

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

Function: Changes the playback rate of the input signal.
Notes: A value of 2.0 results in the output audio playing one octave higher. A value of 0.5, results in the output audio playing one octave lower.

The default value is 1.0. The range of valid values is 0.5 to 2.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.10 renderingAlgorithm as Integer

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

Function: Type of rendering algorithm used.
Notes: Depending on the current output format of the AVAudioEnvironmentNode instance, only a subset of the rendering algorithms may be supported. An array of valid rendering algorithms is retrieved by calling the applicableRenderingAlgorithms function of the AVAudioEnvironmentNode instance.

The default rendering algorithm is AVAudio3DMixingRenderingAlgorithmEqualPowerPanning. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.

The rendering algorithms differ in terms of quality and cpu cost. AVAudio3DMixingRenderingAlgorithmEqualPowerPanning is the simplest panning algorithm and also the least expensive computationally.

If the AVAudioEnvironmentNode instance is set to a multichannel output format, only certain rendering algorithms know how to render to all of the output channels (currently AVAudio3DMixingRenderingAlgorithmSoundField). If any of the other rendering algorithms are selected, the AVAudioEnvironmentNode instance will render to only the first two output channels.

With the exception of AVAudio3DMixingRenderingAlgorithmSoundField, while the mixer is rendering to multi channel hardware, audio data will only be rendered to channels 1 & 2.
(Read and Write property)

5.11.11 reverbBlend as Double

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

Function: Controls the blend of dry and reverb processed audio.
Notes: This property controls the amount of the source‚Äôs audio that will be processed by the reverb by the AVAudioEnvironmentNode instance. A value of 0.5 will result in an equal blend of dry and processed (wet) audio.

The default is 0.0. The range of valid values is 0.0 (completely dry) to 1.0 (completely wet). This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.11.12 volume as Double

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

Function: The buss's input volume.
Notes: The default value is 1.0. The range of valid values are 0.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode and AVAudioMixerNode class mixers.
(Read and Write property)

5.11.13 Constants

Rendering Algorithms

Constant	Value	Description
RenderingAlgorithmEqualPowerPanning	0	Pans the data of the mixer bus into a stereo field. This algorithm is an to the pan knob found on a mixing board channel strip.
RenderingAlgorithmHRTF	2	The Head Related Transfer Function is a high quality algorithm usin ing to emulate 3 dimensional space in headphones. HRTF is a cpu i algorithm.
RenderingAlgorithmHRTFHQ	6	Higher quality HRTF rendering algorithm compared to AVAudio3DMix deringAlgorithmHRTF. Improvements have been made to the overall frequency response and l tion of sources in a 3D space. Available in MacOS 10.13 or newer.
RenderingAlgorithmSoundField	3	Designed for rendering to multi channel hardware. The mixer takes da rendered with SoundField and distributes it amongst all the output c with a weighting toward the location in which the sound derives. It effective for ambient sounds, which may derive from a specific location i yet should be heard through the listener,Äôs entire space.
RenderingAlgorithmSphericalHead	1	Designed to emulate 3 dimensional space in headphones by simulatin aural time delays and other spatial cues. This algorithm is slightly le intensive than AVAudio3DMixingRenderingAlgorithmHRTF.
RenderingAlgorithmStereoPassThrough	5	This algorithm should be used when no localization is desired for th data. Setting this algorithm tells the mixer to take mono/stereo input a it directly to channels 1 & 2 without localization.

5.12 class AVAudioIONodeMBS

5.12.1 class AVAudioIONodeMBS

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

Function: The AVAudioIONode class is the base class for nodes that connects to the system's audio input or output.
Notes: Subclass of the AVAudioNodeMBS class.

5.12.2 Methods

5.12.3 Constructor

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

Function: The constructor.

5.12.4 Properties

5.12.5 audioUnit as Integer

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

Function: The node's underlying audio unit, if any.
Notes: Value is the handle.
(Read only property)

5.12.6 presentationLatency as Double

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

Function: The presentation, or hardware, latency.
Notes: (Read only property)

5.13 class AVAudioMixerNodeMBS

5.13.1 class AVAudioMixerNodeMBS

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

Function: The AVAudioMixerNode class represents a node that mixes its inputs to a single output.
Notes: Mixers may have any number of inputs. The mixer node accepts input at any sample rate and efficiently combines sample rate conversions. It also accepts any channel count and will correctly upmix or downmix to the output channel count.
Subclass of the AVAudioNodeMBS class.

5.13.2 Methods

5.13.3 Constructor

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

Function: The construtor.

5.13.4 Properties

5.13.5 nextAvailableInputBus as Integer

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

Function: Returns an unused input bus.
Notes: This method finds and returns the first input bus to which no other node is connected.
(Read only property)

5.13.6 obstruction as Double

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

Function: Simulates filtering of the direct path of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Only the direct path of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.

(Read and Write property)

5.13.7 occlusion as Double

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

Function: Simulates filtering of the direct and reverb paths of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Both the direct and reverb paths of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.8 pan as Double

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

Function: The buss's stereo pan.
Notes: The default value is 0.0. A value in the range -1.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.9 position as AVAudio3DPointMBS

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

Function: The location of the source in the 3D environment.
Notes: The coordinates are specified in meters. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.10 rate as Double

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

Function: Changes the playback rate of the input signal.
Notes: A value of 2.0 results in the output audio playing one octave higher. A value of 0.5, results in the output audio playing one octave lower.

The default value is 1.0. The range of valid values is 0.5 to 2.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.11 renderingAlgorithm as Integer

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

Function: Type of rendering algorithm used.
Notes: Depending on the current output format of the AVAudioEnvironmentNode instance, only a subset of the rendering algorithms may be supported. An array of valid rendering algorithms is retrieved by calling the applicableRenderingAlgorithms function of the AVAudioEnvironmentNode instance.

The default rendering algorithm is AVAudio3DMixingRenderingAlgorithmEqualPowerPanning. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.12 reverbBlend as Double

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

Function: Controls the blend of dry and reverb processed audio.
Notes: This property controls the amount of the source‚Äôs audio that will be processed by the reverb by the AVAudioEnvironmentNode instance. A value of 0.5 will result in an equal blend of dry and processed (wet) audio.

The default is 0.0. The range of valid values is 0.0 (completely dry) to 1.0 (completely wet). This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.13.13 volume as Double

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

Function: The buss’s input volume.
Notes: The default value is 1.0. The range of valid values are 0.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode and AVAudioMixerNode class mixers.
(Read and Write property)

5.14 class AVAudioNodeMBS

5.14.1 class AVAudioNodeMBS

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

Function: The AVAudioNode class is an abstract class for an audio generation, processing, or I/O block.
Notes: AVAudioEngine objects contain instances of various AVAudioNode subclasses. This base class provides certain common functionality.

Nodes have input and output busses, which can be thought of as connection points. For example, an effect typically has one input bus and one output bus. A mixer typically has multiple input busses and one output bus.

Busses have formats, expressed in terms of sample rate and channel count. When making connections between nodes, often the format must match exactly. However, there are exceptions such as the AVAudioMixerNode and AVAudioOutputNode classes.

Nodes do not currently provide useful functionality until attached to an engine.
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
- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.3

5.14.2 Methods

5.14.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.14.4 Constructor

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

Function: The private constructor.

5.14.5 inputFormatForBus(busIndex as Integer) as AVAudioFormatMBS

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

Function: Returns the input format for the specified bus.
Notes: Returns an AVAudioFormatMBS object representing the input format of the bus.

5.14.6 installTapOnBus(busIndex as Integer, bufferSize as UInt32 = 0, format as AVAudioFormatMBS = nil, tag as Variant = nil)

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

Function: Installs an audio tap on the bus to record, monitor, and observe the output of the node.
Notes: bus: The node output bus to which to attach the tap.
bufferSize: The requested size of the incoming buffers. The implementation may choose another size.
format: If non-nil, attempts to apply this as the format of the specified output bus.

This should only be done when attaching to an output bus which is not connected to another node; an error will result otherwise. The tap and connection formats (if non-nil) on the specified bus should be identical. Otherwise, the latter operation will override any previously set format.

For AVAudioOutputNode, tap format must be specified as nil.

The plugin calls the Tap event with the data periodically.

Only one tap may be installed on any bus. Taps may be safely installed and removed while the engine is running.

5.14.7 nameForInputBus(busIndex as Integer) as string

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

Function: The name of an input bus.

5.14.8 nameForOutputBus(busIndex as Integer) as string

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

Function: The name of the output bus.

5.14.9 outputFormatForBus(busIndex as Integer) as AVAudioFormatMBS

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

Function: Returns the output format for the specified bus.
Notes: Returns an AVAudioFormatMBS object representing the input format of the bus.

5.14.10 removeTapOnBus(busIndex as Integer)

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

Function: Removes an audio tap on a bus.
Notes: bus: The node output bus whose tap is to be removed.

5.14.11 reset

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

Function: Clear a unit's previous processing state.

5.14.12 Properties

5.14.13 AUAudioUnitHandle as Integer

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: An AUAudioUnit wrapping or underlying the implementation's AudioUnit.
Notes: This provides an AUAudioUnit which either wraps or underlies the implementation's AudioUnit, depending on how that audio unit is packaged. Applications can interact with this AUAudioUnit to control custom properties, select presets, change parameters, etc.

No operations that may conflict with state maintained by the engine should be performed directly on the audio unit. These include changing initialization state, stream formats, channel layouts or connections to other audio units.

Available on MacOS 10.13 or newer.
(Read only property)

5.14.14 engine as AVAudioEngineMBS

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

Function: The audio engine of the node.
Notes: Returns nil if the node is not attached to an audio engine.
(Read only property)

5.14.15 Handle as Integer

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

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

5.14.16 lastRenderTime as AVAudioTimeMBS

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

Function: The time for which the node most recently rendered.
Notes: Return nil if the engine is not running or if the node is not connected to an input or output node.
(Read only property)

5.14.17 latency as Double

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The processing latency of the node, in seconds.
Notes: This property reflects the delay between when an impulse in the audio stream arrives at the input vs. output of the node. This should reflect the delay due to signal processing (e.g. filters, FFT's, etc.), not delay or reverberation which is being applied as an effect. A value of zero indicates either no latency or an unknown latency.

Available on MacOS 10.13 or newer.
(Read only property)

5.14.18 numberOfInputs as Integer

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

Function: The number of input busses for the node.
Notes: (Read only property)

5.14.19 numberOfOutputs as Integer

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

Function: The number of output busses for the node.
Notes: (Read only property)

5.14.20 outputPresentationLatency as Double

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The maximum render pipeline latency downstream of the node, in seconds.
Notes: This describes the maximum time it will take for the audio at the output of a node to be presented. For instance, the output presentation latency of the output node in the engine is:

- zero in manual rendering mode
- the presentation latency of the device itself when rendering to an audio device (see AVAudioIONodeMBS.presentationLatency)

The output presentation latency of a node connected directly to the output node is the output node's presentation latency plus the output node's processing latency (see latency).

For a node which is exclusively in the input node chain (i.e. not connected to engine's
output node), this property reflects the latency for the output of this node to be
presented at the output of the terminating node in the input chain.

A value of zero indicates either an unknown or no latency.

Note that this latency value can change as the engine is reconfigured (started/stopped,
connections made/altered downstream of this node etc.). So it is recommended not to cache
this value and fetch it whenever it's needed.

Available on MacOS 10.13 or newer.
(Read only property)

5.14.21 Events

5.14.22 Tap(bus as Integer, bufferSize as UInt32, format as AVAudioFormatMBS, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, tag as Variant)

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

Function: The tap event.
Notes: Called after using installTapOnBus periodically.

5.15 class AVAudioOutputNodeMBS

5.15.1 class AVAudioOutputNodeMBS

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

Function: The AVAudioOutputNode class represents a audio node that connects to the system's audio output.
Notes: This audio node has one element. The format of the output scope reflects the audio hardware sample rate and channel count. The format of the input scope is initially the same as that of the output, but you may set it to a different format, in which case the audio node will convert.
Subclass of the AVAudioIONodeMBS 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 20.0pr5

5.15.2 Methods

5.15.3 Constructor

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

Function: The private constructor.

5.15.4 Properties

5.15.5 CurrentDeviceID as UInt32

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

Function: Get or set the current device.
Notes: Please check OutputDevices dictionary for which devices are available.
(Read and Write property)

5.15.6 DefaultDeviceID as UInt32

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the Core Audio ID of the default device.
Notes: (Read only property)

5.15.7 OutputDevices as Dictionary

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries a dictionary with output devices.
Example:

```
Dim e As New AVAudioEngineMBS
Dim a As AVAudioOutputNodeMBS = e.outputNode
Dim d As Dictionary = a.OutputDevices

For Each key As Variant In d.keys
dim name as string = d.Value(key)

MsgBox key+": "+name
Next
```

Notes: Key is the ID and value is the name of the device.
(Read only property)

5.16 class AVAudioPCMBufferMBS

5.16.1 class AVAudioPCMBufferMBS

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

Function: The AVAudioPCMBuffer class is a subclass of AVAudioBuffer for use with PCM audio formats.
Notes: The PCM buffer class also provides methods useful for manipulating buffers of audio in PCM format.
Subclass of the AVAudioBufferMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 20.3
- Write audio file with samples using AVFoundation

5.16.2 Methods

5.16.3 Constructor(format as AVAudioFormatMBS, frameCapacity as Integer)

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

Function: Initializes a newly allocated PCM audio buffer instance for PCM audio data.
Notes: format: The format of the PCM audio to be contained in the buffer.
frameCapacity: The capacity of the buffer in PCM sample frames.

5.16.4 floatChannelDataCopy(ChannelIndex as Integer) as Memoryblock

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

Function: Copies buffer's audio samples as floating point values.
Notes: Returns memoryblock with samples only for the given channel.
Handles stride automatically for you.

5.16.5 int16ChannelDataCopy(ChannelIndex as Integer) as Memoryblock

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

Function: Copies buffer's audio samples as 16-bit integer point values.
Notes: Returns memoryblock with samples only for the given channel.
Handles stride automatically for you.

5.16.6 int32ChannelDataCopy(ChannelIndex as Integer) as Memoryblock

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

Function: Copies buffer's audio samples as 32-bit integer point values.
Notes: Returns memoryblock with samples only for the given channel.
Handles stride automatically for you.

5.16.7 level(ChannelIndex as Integer) as Double

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

Function: Calculates level for a channel.
Notes: Value returned is between around 0.0 (nothing) to 1.0 (maximum).
Negative values are for errors.

5.16.8 setFloatChannelData(ChannelIndex as Integer, Data as Memoryblock) as boolean

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Writes channel to the buffer.
Notes: Please set frameLength to a matching value.
Data.size should be frameLength * 4.

Returns true on success or false on failure.
Handles stride automatically for you.

5.16.9 setInt16ChannelData(ChannelIndex as Integer, Data as Memoryblock) as boolean

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Writes channel to the buffer.
Notes: Please set frameLength to a matching value.
Data.size should be frameLength * 2.

Returns true on success or false on failure.
Handles stride automatically for you.

5.16.10 setInt32ChannelData(ChannelIndex as Integer, Data as Memoryblock) as boolean

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Writes channel to the buffer.
Notes: Please set frameLength to a matching value.
Data.size should be frameLength * 4.

Returns true on success or false on failure.
Handles stride automatically for you.

5.16.11 Properties

5.16.12 floatChannelData as Ptr

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

Function: The buffer's audio samples as floating point values.
Notes: The floatChannelData property returns pointers to the buffer,Äôs audio samples, if the buffer,Äôs format is 32-bit float. It returns nil if it is another format.

The returned pointer is to format.channelCount pointers to float. Each of these pointers is to frameLength valid samples, which are spaced by stride samples.

If the format is not interleaved, as with the standard deinterleaved float format, then the pointers will be to separate chunks of memory and the stride property value is 1.

If the format is interleaved, then the pointers will refer to the same buffer of interleaved samples, each offset by 1 frame, and the stride property value is the number of interleaved channels.
(Read only property)

5.16.13 frameCapacity as Integer

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

Function: The buffer's capacity, in audio sample frames.
Notes: (Read only property)

5.16.14 frameLength as Integer

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

Function: The current number of valid sample frames in the buffer.
Notes: You may modify the length of the buffer as part of an operation that modifies its contents. The length must be less than or equal to the frameCapacity. Modifying frameLength will update the mDataByteSize field in each of the underlying AudioBufferList struct‚Äôs AudioBuffer struct correspondingly, and vice versa. Note that in the case of deinterleaved formats, the mDataByteSize field will refers the size of one channel‚Äôs worth of audio samples.

The length must be less than or equal to the frameCapacity. In the case of deinterleaved formats, frameCapacity will refer to the size of one channel‚Äôs worth of audio samples.
(Read and Write property)

5.16.15 int16ChannelData as Ptr

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

Function: Access the buffer's 16bit integer audio samples.
Notes: The int16ChannelData property returns the buffer's audio samples if the buffer's format has 2-byte integer samples, or nil if it is another format.

See floatChannelData for more information.
(Read only property)

5.16.16 int32ChannelData as Ptr

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

Function: Access the buffer's 32-bit integer audio samples.
Notes: The int32ChannelData property returns the buffer's audio samples if the buffer's format has 4-byte integer samples, or nil if it is another format.

See floatChannelData for more information.
(Read only property)

5.16.17 stride as Integer

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

Function: The buffer's number of interleaved channels.
Notes: (Read only property)

5.17 class AVAudioPlayerNodeMBS

5.17.1 class AVAudioPlayerNodeMBS

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

Function: The AVAudioPlayerNode class plays buffers or segments of audio files.
Notes: AVAudioPlayerNode supports scheduling the playback of AVAudioBuffer instances, or segments of audio files opened via AVAudioFile. Buffers and segments may be scheduled at specific points in time, or to play immediately following preceding segments.

Normally, you will want to configure the node‚Äôs output format with the same number of channels as are in the files and buffers to be played. Otherwise, channels will be dropped or added as required. It is usually better to use an AVAudioMixerNode to do this.

Similarly, when playing file segments, the node will sample rate convert if necessary, but it is often preferable to configure the node‚Äôs output sample rate to match that of the file(s) and use a mixer to perform the rate conversion.

When playing buffers, there is an implicit assumption that the buffers are at the same sample rate as the node‚Äôs output format.

This class overrides the AVAudioNode method reset and unschedules all previously scheduled buffers and file segments, and also returns the player timeline to sample time 0.

Player Timeline
The usual AVAudioNode sample times (as observed by lastRenderTime) have an arbitrary zero point. AVAudioPlayerNode superimposes a second ‚Äúplayer timeline‚Äù on top of this, to reflect when the player was started, and intervals during which it was paused. The methods nodeTimeForPlayerTime and playerTimeForNodeTime convert between the two.

Scheduling Playback Time
The scheduleBuffer, scheduleFile, and scheduleSegment methods take an AVAudioTime when parameter. This is interpreted as follows:

- If the when parameter is nil:
  - If there have been previous commands, the new one is played immediately following the last one.
  - Otherwise, if the node is playing, the event is played in the very near future.
  - Otherwise, the command is played at sample time 0.
- If the when parameter is a sample time, the parameter is interpreted as such.
- If the when parameter is a host time it is ignored unless sample time not valid.

The scheduling methods will fail if:

- A buffer's channel count does not match that of the node's output format.
- A file can't be accessed.
- An AVAudioTime specifies neither a valid sample time or host time.
- A segment's start frame or frame count is negative.

Subclass of the AVAudioNodeMBS class.
Blog Entries

- MBS Xojo Plugins, version 18.4pr4

5.17.2 Methods

5.17.3 Constructor

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

Function: The constructor.

5.17.4 nodeTimeForPlayerTime(playerTime as AVAudioTimeMBS) as AVAudioTimeMBS

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

Function: Convert from player time to node time.
Notes: playerTime: A time relative to the player,Äôs start time.

Returns a node time.

This method and its inverse playerTimeForNodeTime are discussed in The Player Timeline. If the player is not playing when this method is called, nil is returned.

5.17.5 pause

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

Function: Pause playback.
Notes: The player's sample time does not advance while the node is paused.

5.17.6 play

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

Function: Start or resume playback immediately.
Notes: This is equivalent to playAtTime with a value of nil.

5.17.7 playAtTime(time as AVAudioTimeMBS = nil)

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

Function: Start or resume playback at a specific time.
Example:

```
Dim p As AVAudioPlayerNodeMBS // your player node
Dim t As AVAudioTimeMBS = AVAudioTimeMBS.timeWithSampleTime(60000, 1000) // 60s

p.playAtTime(t)
```

Notes: when: The node time at which to start or resume playback. Passing nil starts playback immediately.

This node is initially paused. Requests to play buffers or file segments are enqueued, and any necessary decoding begins immediately. Playback does not begin, however, until the player has started playing, via this method.

5.17.8 playerTimeForNodeTime(nodeTime as AVAudioTimeMBS) as AVAudioTimeMBS

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

Function: Convert from node time to player time.
Notes: nodeTime: The node time.

Returns a time relative to the player,Äôs start time.

This method and its inverse nodeTimeForPlayerTime are discussed in The Player Timeline. If the player is not playing when this method is called, nil is returned.

5.17.9 prepareWithFrameCount(frameCount as UInt32)

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

Function: Prepares previously scheduled file regions or buffers for playback.
Notes: frameCount: The number of sample frames of data to be prepared before returning.

5.17.10 scheduleBuffer(buffer as AVAudioPCMBufferMBS, tag as Variant = nil)

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

Function: Schedule playing samples from an audio buffer.
Notes: buffer: The buffer to play.

Calls scheduleBufferCompleted event later when playback is done.
time will be nil and options be zero in that case.

Schedules the buffer to be played following any previously scheduled commands.
See also:

- 5.17.11 scheduleBuffer(buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 820
- 5.17.12 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 821
- 5.17.13 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 821

5.17.11 scheduleBuffer(buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil)

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

Function: Schedule playing samples from an audio buffer at the specified time and with the specified playback options.
Notes: buffer: The buffer to play.
when: The time at which to play the buffer.
options: Playback options.

Calls later scheduleBufferCompleted event after the buffer has completely played or the player is stopped.
See also:

- 5.17.10 scheduleBuffer(buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 820

- 5.17.12 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 821
- 5.17.13 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 821

5.17.12 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, tag as Variant = nil)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Schedule playing samples from an AVAudioBuffer.
Notes: buffer: the buffer to play
callbackType: option to specify when the completion handler must be called

The scheduleBufferCompleted event is called after the buffer has been consumed by the player or has finished playing back or the player is stopped.

Schedules the buffer to be played following any previously scheduled commands.

Available in MacOS 10.13 or newer.
See also:

- 5.17.10 scheduleBuffer(buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 820
- 5.17.11 scheduleBuffer(buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 820
- 5.17.13 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 821

5.17.13 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Schedule playing samples from an AVAudioBuffer.
Notes: buffer: the buffer to play
time: the time at which to play the buffer. see the discussion of timestamps, above.
options: options for looping, interrupting other buffers, etc.
callbackType: option to specify when the completion handler must be called.

The scheduleBufferCompleted event is called after the buffer has been consumed by the player or has finished playing back or the player is stopped.

Available in MacOS 10.13 or newer.
See also:

- 5.17.10 scheduleBuffer(buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 820
- 5.17.11 scheduleBuffer(buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant = nil) 820
- 5.17.12 scheduleBuffer(callbackType as Integer, buffer as AVAudioPCMBufferMBS, tag as Variant = nil) 821

5.17.14 scheduleFile(callbackType as Integer, file as AVAudioFileMBS, time as AVAudioTimeMBS, tag as Variant = nil)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Schedule playing of an entire audio file.
Notes: file: the file to play
time: the time at which to play the file. see the discussion of timestamps, above.
callbackType: option to specify when the completion handler must be called

The scheduleFileCompleted event is called after the file has been consumed by the player or has finished playing back or the player is stopped.

Available in MacOS 10.13 or newer.
See also:

- 5.17.15 scheduleFile(file as AVAudioFileMBS, time as AVAudioTimeMBS, tag as Variant = nil) 822

5.17.15 scheduleFile(file as AVAudioFileMBS, time as AVAudioTimeMBS, tag as Variant = nil)

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

Function: Schedule playing of an entire audio file.
Notes: file: The file to play.
when: The time at which to play the buffer.

Calls later scheduleFileCompleted event when playback is done.
See also:

- 5.17.14 scheduleFile(callbackType as Integer, file as AVAudioFileMBS, time as AVAudioTimeMBS, tag as Variant = nil) 822

5.17.16 scheduleSegment(callbackType as Integer, file as AVAudioFileMBS, time as AVAudioTimeMBS, startFrame as Int64, frameCount as Int64, tag as Variant = nil)

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Schedule playing a segment of an audio file.
Notes: file: the file to play
startFrame: the starting frame position in the stream
frameCount: the number of frames to play
time: the time at which to play the region. see the discussion of timestamps, above.
callbackType: option to specify when the completion handler must be called

The scheduleSegmentCompleted event is called after the segment has been consumed by the player or has finished playing back or the player is stopped.

Available in MacOS 10.13 or newer.
See also:

- 5.17.17 scheduleSegment(file as AVAudioFileMBS, time as AVAudioTimeMBS, startFrame as Int64, frameCount as Int64, tag as Variant = nil) 823

5.17.17 scheduleSegment(file as AVAudioFileMBS, time as AVAudioTimeMBS, startFrame as Int64, frameCount as Int64, tag as Variant = nil)

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

Function: Schedule playing a segment of an audio file.
Notes: file: The audio file of the file to play.
startFrame: The starting frame position in the stream.
frameCount: The number of frames to play.
when: The time at which to play the buffer.

Calls later scheduleSegmentCompleted event.
See also:

- 5.17.16 scheduleSegment(callbackType as Integer, file as AVAudioFileMBS, time as AVAudioTimeMBS, startFrame as Int64, frameCount as Int64, tag as Variant = nil) 823

5.17.18 stop

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

Function: Clear all of the node's previously scheduled events and stop playback.
Notes: All of the node,Äôs previously scheduled events are cleared, including any that are in the middle

of playing. The node,Äôs sample time (and therefore the times to which new events are to be scheduled) is reset to 0, and will not proceed until the node is started again by sending a play or playAtTime message.

5.17.19 Properties

5.17.20 obstruction as Double

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

Function: Simulates filtering of the direct path of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Only the direct path of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.21 occlusion as Double

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

Function: Simulates filtering of the direct and reverb paths of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Both the direct and reverb paths of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.22 pan as Double

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

Function: The buss's stereo pan.
Notes: The default value is 0.0. A value in the range -1.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.23 Playing as Boolean

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

Function: Indicates whether or not the player is playing.
Notes: (Read only property)

5.17.24 position as AVAudio3DPointMBS

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

Function: The location of the source in the 3D environment.
Notes: The coordinates are specified in meters. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.25 rate as Double

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

Function: Changes the playback rate of the input signal.
Notes: A value of 2.0 results in the output audio playing one octave higher. A value of 0.5, results in the output audio playing one octave lower.

The default value is 1.0. The range of valid values is 0.5 to 2.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.26 renderingAlgorithm as Integer

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

Function: Type of rendering algorithm used.
Notes: Depending on the current output format of the AVAudioEnvironmentNode instance, only a subset of the rendering algorithms may be supported. An array of valid rendering algorithms is retrieved by calling the applicableRenderingAlgorithms function of the AVAudioEnvironmentNode instance.

The default rendering algorithm is AVAudio3DMixingRenderingAlgorithmEqualPowerPanning. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.27 reverbBlend as Double

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

Function: Controls the blend of dry and reverb processed audio.
Notes: This property controls the amount of the source‚Äôs audio that will be processed by the reverb by the AVAudioEnvironmentNode instance. A value of 0.5 will result in an equal blend of dry and processed (wet) audio.

The default is 0.0. The range of valid values is 0.0 (completely dry) to 1.0 (completely wet). This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.17.28 volume as Double

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

Function: The buss's input volume.
Notes: The default value is 1.0. The range of valid values are 0.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode and AVAudioMixerNode class mixers.
(Read and Write property)

5.17.29 Events

5.17.30 scheduleBufferCompleted(callbackType as Integer, buffer as AVAudioPCMBufferMBS, time as AVAudioTimeMBS, options as Integer, tag as Variant)

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

Function: Called after the buffer has completely played or the player is stopped.
Notes: callbackType: The requested callback type or -1 for older APIs.

5.17.31 scheduleFileCompleted(callbackType as Integer, file as AVAudioFileMBS, time as AVAudioTimeMBS, tag as Variant)

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

Function: Called after the file has completely played or the player is stopped.
Notes: callbackType: The requested callback type or -1 for older APIs.

5.17.32 scheduleSegmentCompleted(callbackType as Integer, file as AVAudioFileMBS, startFrame as Int64, frameCount as Int64, time as AVAudioTimeMBS, tag as Variant)

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

Function: Called after the segment has completely played or the player is stopped.
Notes: callbackType: The requested callback type or -1 for older APIs.

5.17.33 Constants

Options controlling buffer scheduling

Constant	Value	Description
BufferInterrupts	2	The buffer interrupts any buffer already playing.
BufferInterruptsAtLoop	4	The buffer interrupts any buffer already playing, at its loop point.
BufferLoops	1	The buffer loops indefinitely.

Completion Handler Modes

Constant	Value	Description
CompletionDataConsumed	0	The buffer or file data has been consumed by the player.
CompletionDataPlayedBack	2	Applicable only when the engine is rendering to/from an audio device. The buffer or file has finished playing. This accounts for both (small) signal processing latencies downstream of the player in the engine, as well as (possibly significant) latency in the audio playback device.
CompletionDataRendered	1	The buffer or file data has been rendered (i.e. output) by the player. This does not account for any signal processing latencies downstream of the player in the engine (see AVAudioNodeMBS.outputPresentationLatency).

5.18 class AVAudioTimeMBS

5.18.1 class AVAudioTimeMBS

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

Function: The AVAudioTime class is used by AVAudioEngine to represent time.
Example:

```
Dim t As AVAudioTimeMBS = AVAudioTimeMBS.timeWithSampleTime(60000, 1000) // 60s
```

Notes: Instances of the class are immutable.
A single moment in time may be represented in two different ways:

- Using mach_absolute_time(), the system's basic clock. Referred to as ‚Äúhost time.‚Äù
- Audio samples at a particular sample rate.

A single AVAudioTime instance may contain either or both representations; it might represent only a sample time, only a host time, or both.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.18.2 Methods

5.18.3 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.18.4 Constructor

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

Function: The private constructor.

5.18.5 extrapolateTimeFromAnchor(anchorTime as AVAudioTimeMBS) as AVAudioTimeMBS

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

Function: Creates an audio time object by converting between host and sample time.
Notes: anchorTime: An AVAudioTime instance with a more complete AudioTimeStamp than that of the receiver.

Returns a newly created and initialized AVAudioTime instance with the extrapolated time.

If anchorTime is an AVAudioTime instance where both host time and sample time are valid, and the receiver is another timestamp where only one of the two is valid, this method returns a new AVAudioTime instance copied from the receiver and where any additional valid fields provided by the anchor are also valid.

5.18.6 hostTimeForSeconds(Seconds as Double) as UInt64

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

Function: Converts seconds to host time.
Notes: seconds: The number of seconds.
Returns the seconds represented as host time.

5.18.7 secondsForHostTime(HostTime as UInt64) as Double

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

Function: Converts host time to seconds.
Notes: hostTime: The host time.
Returns the host time represented as seconds.

5.18.8 timeWithHostTime(HostTime as UInt64) as AVAudioTimeMBS

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

Function: Creates an audio time object with the specified host time.
Notes: hostTime: The host time.
Returns a newly created and initialized AVAudioTime instance with the extrapolated time.
See also:

- 5.18.9 timeWithHostTime(hostTime as UInt64, SampleTime as Int64, sampleRate as Double) as AVAudioTimeMBS 830

5.18.9 timeWithHostTime(hostTime as UInt64, SampleTime as Int64, sampleRate as Double) as AVAudioTimeMBS

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

Function: Creates an audio time object with the specified host time, sample time, and sample rate.
Notes: hostTime: The host time.
sampleTime: The sample time.
sampleRate: The sample rate.

Returns a newly created and initialized AVAudioTime instance with the specified host time, sample time, and sample rate.
See also:

- 5.18.8 timeWithHostTime(HostTime as UInt64) as AVAudioTimeMBS 829

5.18.10 timeWithSampleTime(SampleTime as Int64, sampleRate as Double) as AVAudioTimeMBS

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

Function: Creates an audio time object with the specified sample time and sample rate.
Example:

```
Dim t As AVAudioTimeMBS = AVAudioTimeMBS.timeWithSampleTime(60000, 1000) // 60s
```

Notes: sampleTime: The sample time.
sampleRate: The sample rate.

Returns a newly created and initialized AVAudioTime instance with the specified sample time and sample rate.

5.18.11 Properties

5.18.12 Handle as Integer

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

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

5.18.13 hostTime as UInt64

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

Function: The host time.
Notes: (Read only property)

5.18.14 hostTimeValid as Boolean

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

Function: Whether the host time value is valid.
Notes: True if the hostTime property is valid, otherwise false.
(Read only property)

5.18.15 sampleRate as Double

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

Function: The sample rate at which the sample time is being expressed.
Notes: (Read only property)

5.18.16 sampleTime as Int64

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

Function: The time as a number of audio samples, as tracked by the current audio device.
Notes: (Read only property)

5.18.17 sampleTimeValid as Boolean

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

Function: Whether the sample time and sample rate properties are valid.
Notes: (Read only property)

5.19 class AVAudioUnitComponentManagerMBS

5.19.1 class AVAudioUnitComponentManagerMBS

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

Function: The AVAudioUnitComponentManager class is a singleton object that provides a way to find audio components that are registered with the system.
Notes: It provides methods to search and query various information about the audio components without opening them. Currently, only audio components that are audio units can only be searched.

The class also supports predefined system tags and arbitrary user tags. Each audio unit can be tagged as part of its definition. AudioUnit Hosts such as Logic or GarageBand can present groupings of audio units based on the tags.

Searching for audio units can be done in various ways:

- Using a NSPredicate instance that contains search strings for tags or descriptions
- Using a block to match on a custom criteria.
- Using an AudioComponentDescription.

Available in OS X v10.10 and later.

5.19.2 Methods

5.19.3 allComponents as AVAudioUnitComponentMBS()

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

Function: Finds all components.
Example:

```
// Needs OS X 10.10 or newer
dim m as new AVAudioUnitComponentManagerMBS

dim a() as AVAudioUnitComponentMBS = m.allComponents

for each c as AVAudioUnitComponentMBS in a
dim d as AVAudioComponentDescriptionMBS = c.audioComponentDescription
'List.AddRow c.Name, c.ManufacturerName, c.LocalizedTypeName, c.VersionString, d.componentType+" "+d.componentSubType
```

```
// we look for the mixer
// AUMixer Apple Mixer 1.6.0 aumx smxr

if d.componentType = "aumx" and d.componentSubType = "smxr" then
dim aa as new AVAudioUnitMBS(d)
MsgBox aa.Name
end if

next
```

5.19.4 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.19.5 AVAudioUnitComponentTagsDidChangeNotification as String

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

Function: The notification name for the notification sent when tags changed.

5.19.6 componentsMatchingDescription(Description as AVAudioComponentDescriptionMBS) as AVAudioUnitComponentMBS()

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

Function: An array of audio component objects that matches the description.
Notes: description: The AudioComponentDescription struct to match. The type, subtype and manufacturer fields are used to search for matching audio units. A value of "" or 0 for any of these fields is a wildcard and returns the first match found.

5.19.7 componentsPassingTest(tag as Variant = nil) as AVAudioUnitComponentMBS()

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

Function: An array of audio components that pass the Test event.
Notes: For each AudioComponent found by the manager, the Test event will be called. If the block returns

true then the AVAudioComponent instance is added to the array.

5.19.8 Constructor

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

Function: The constructor.

5.19.9 Destructor

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

Function: The destructor.

5.19.10 sharedAudioUnitComponentManager as AVAudioUnitComponentManagerMBS

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

Function: Returns the shared component manager.

5.19.11 standardLocalizedTagNames as string()

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

Function: An array of the the localized standard system tags defined by the audio units.

5.19.12 tagNames as string()

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

Function: An array of all tags associated with the current user, as well as all system tags defined by the audio units.

5.19.13 Properties

5.19.14 Handle as Integer

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

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

5.19.15 Events

5.19.16 TagsDidChange(notification as NSNotificationMBS)

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

Function: The event called when tags changed.

5.19.17 Test(component as AVAudioUnitComponentMBS, byref stop as Boolean, tag as Variant) as Boolean

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

Function: The event to apply to the audio unit components.
Notes: Component: The component to test.
stop: A reference to a Boolean value. The block can set the value to true to stop further processing of the search. The stop argument is an out-only argument. You should only ever set this Boolean to true within the event.

The event returns a Boolean value that indicates whether comp passed the test.

5.20 class AVAudioUnitComponentMBS

5.20.1 class AVAudioUnitComponentMBS

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

Function: The AVAudioUnitComponent class provides details about an audio unit such as: type, subtype, manufacturer, and location.
Example:

```
// Needs OS X 10.10 or newer
dim m as new AVAudioUnitComponentManagerMBS

dim a() as AVAudioUnitComponentMBS = m.allComponents

for each c as AVAudioUnitComponentMBS in a
dim d as AVAudioComponentDescriptionMBS = c.audioComponentDescription
'List.AddRow c.Name, c.ManufacturerName, c.LocalizedTypeName, c.VersionString, d.componentType+"
"+d.componentSubType

// we look for the mixer
// AUMixer Apple Mixer 1.6.0 aumx smxr

if d.componentType = "aumx" and d.componentSubType = "smxr" then
dim aa as new AVAudioUnitMBS(d)
MsgBox aa.Name
end if

next
```

Notes: User tags can be added to the AVAudioUnitComponent which can be queried later for display.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.20.2 Methods

5.20.3 allTagNames as string()

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

Function: An array of tag names for the audio unit component.

5.20.4 available as boolean

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

Function: Whether this class is available for use.
Notes: Should return true on OS X 10.10 and newer.

5.20.5 Constructor

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

Function: The private constructor.

5.20.6 SetUserTagNames(tags() as string)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets an array of tags created by the user.

5.20.7 supportsNumberOfChannels(numInputChannels as Integer, numOutputChannels as Integer) as boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns whether the audio unit component supports the specified number of input and output channels.
Notes: numInputChannels: The number of input channels.
numOutputChannels: The number of output channels.

Returns true if the audio unit component supports the specified number of input and output channels, otherwise false.

5.20.8 userTagNames as string()

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: An array of tags created by the user.

5.20.9 Properties

5.20.10 audioComponentDescription as AVAudioComponentDescriptionMBS

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

Function: The AudioComponentDescription of the audio unit component.
Notes: (Read only property)

5.20.11 componentFile as FolderItem

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The folderitem of the audio unit component.
Notes: (Read only property)

5.20.12 componentURL as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The URL of the audio unit component.
Notes: (Read only property)

5.20.13 configurationDictionary as Dictionary

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The audio unit component,Äôs configuration dictionary.
Notes: (Read only property)

5.20.14 Handle as Integer

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

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

5.20.15 hasCustomView as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the audio unit component has a custom view.
Notes: True if the component has a custom view, otherwise false.
(Read only property)

5.20.16 hasMIDIInput as Boolean

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

Function: Whether the audio unit component has midi input.
Notes: (Read only property)

5.20.17 hasMIDIOutput as Boolean

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

Function: Whether the audio unit component has midi output.
Notes: (Read only property)

5.20.18 iconFile as FolderItem

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The folderitem of an icon the represents the audio unit component.
Notes: (Read only property)

5.20.19 iconURL as String

Plugin Version: 15.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The URL of an icon the represents the audio unit component.
Notes: (Read only property)

5.20.20 LocalizedTypeName as String

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

Function: The localized type name of the component.
Notes: (Read only property)

5.20.21 ManufacturerName as String

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

Function: The name of the manufacturer of the audio unit component.
Notes: (Read only property)

5.20.22 Name as String

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

Function: The name of the audio unit component.
Notes: (Read only property)

5.20.23 passesAUVal as Boolean

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

Function: Whether the audio unit component passes the AUVal.
Notes: True if the component passes the AUVal, otherwise false.
(Read only property)

5.20.24 SandboxSafe as Boolean

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

Function: Whether the audio unit component is safe for sandboxing.
Notes: True if the component is safe for sandboxing, otherwise false. This only applies to the current process.
(Read only property)

5.20.25 TypeName as String

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

Function: The audio unit component type.
Notes: (Read only property)

5.20.26 Version as Integer

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

Function: The audio unit component version number.
Notes: The version number is an NSNumber comprised of a hexadecimal number with major, minor, and a dot-release format: 0xMMMMmmDD.
(Read only property)

5.20.27 VersionString as String

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

Function: A string representing the audio unit component version number.
Notes: (Read only property)

5.21 class AVAudioUnitDelayMBS

5.21.1 class AVAudioUnitDelayMBS

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

Function: The AVAudioUnitDelay class is an AVAudioUnitEffect subclass that implements a delay effect.
Notes: A delay unit delays the input signal by the specified time interval and then blends it with the input signal. The amount of high frequency roll-off can also be controlled in order to simulate the effect of a tape delay.
Subclass of the AVAudioUnitEffectMBS class.

5.21.2 Methods

5.21.3 Constructor

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

Function: The constructor.

5.21.4 Properties

5.21.5 delayTime as Double

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

Function: The time taken by the delayed input signal to reach the output.
Notes: The delay is specified in seconds. The default value is 1. The valid range of values is 0 to 2 seconds.
(Read and Write property)

5.21.6 feedback as Double

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

Function: The amount of the output signal fed back into the delay line.
Notes: The feedback is specified as a percentage. The default value is 50%. The valid range of values is -100% to 100%.
(Read and Write property)

5.21.7 lowPassCutoff as Double

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

Function: The cutoff frequency, in Hz, above which high frequency content is rolled off.
Notes: The default value is 15000 Hz. The valid range of values is 10 Hz through (sampleRate/2).
(Read and Write property)

5.21.8 wetDryMix as Double

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

Function: The blend of the wet and dry signals.
Notes: The blend is specified as a percentage. The default value is 100%. The valid range of values is 0% (all dry) through 100% (all wet).
(Read and Write property)

5.22 class AVAudioUnitDistortionMBS

5.22.1 class AVAudioUnitDistortionMBS

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

Function: The AVAudioUnitDistortion class is an AVAudioUnitEffect subclass that implements a multistage distortion effect.
Notes: Subclass of the AVAudioUnitEffectMBS class.

5.22.2 Methods

5.22.3 Constructor

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

Function: The constructor.

5.22.4 loadFactoryPreset(preset as Integer)

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

Function: Configures the audio distortion unit by loading a distortion preset.
Notes: preset: The distortion preset. See AVAudioUnitDistortionPreset for possible values.

5.22.5 Properties

5.22.6 preGain as Double

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

Function: The gain, in decibels, applied to the signal before being distorted.
Notes: The default value is -6 db. The valid range of values is -80 db to 20 db.
(Read and Write property)

5.22.7 wetDryMix as Double

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

Function: The blend of the distorted and dry signals.
Notes: The blend is specified as a percentage. The default value is 50%. The range is 0% (all dry) through 100% (all wet).
(Read and Write property)

5.22.8 Constants

Preset Audio Distortions

Constant	Value	Description
PresetDrumsBitBrush	0	Preset that provides a ,ÄúDrumsBitBrush,Äù distortion.
PresetDrumsBufferBeats	1	Preset that provides a ,ÄúDrumsBuffersBitBrush,Äù distortion.
PresetDrumsLoFi	2	Preset that provides a ,ÄúDrumsLoFi,Äù distortion.
PresetMultiBrokenSpeaker	3	Preset that provides a ,ÄúMultiBrokenSpeaker,Äù distortion.
PresetMultiCellphoneConcert	4	Preset that provides a ,ÄúMultiCellphoneConcert,Äù distortion.
PresetMultiDecimated1	5	Preset that provides a ,ÄúMultiDecimated1,Äù distortion.
PresetMultiDecimated2	6	Preset that provides a ,ÄúMultiDecimated2,Äù distortion.
PresetMultiDecimated3	7	Preset that provides a ,ÄúMultiDecimated3,Äù distortion.
PresetMultiDecimated4	8	Preset that provides a ,ÄúMultiDecimated4,Äù distortion.
PresetMultiDistortedCubed	10	Preset that provides a ,ÄúMultiDistortedCubed,Äù distortion.
PresetMultiDistortedFunk	9	Preset that provides a ,ÄúMultiDistortedFunk,Äù distortion.
PresetMultiDistortedSquared	11	Preset that provides a ,ÄúMultiDistortedSquared,Äù distortion.
PresetMultiEcho1	12	Preset that provides a ,ÄúMultiEcho1,Äù distortion.
PresetMultiEcho2	13	Preset that provides a ,ÄúMultiEcho2,Äù distortion.
PresetMultiEchoTight1	14	Preset that provides a ,ÄúMultiEchoTight1,Äù distortion.
PresetMultiEchoTight2	15	Preset that provides a ,ÄúMultiEchoTight2,Äù distortion.
PresetMultiEverythingIsBroken	16	Preset that provides a ,ÄúMultiEverythingIsBroken,Äù distortion.
PresetSpeechAlienChatter	17	Preset that provides a ,ÄúSpeechAlienChatter,Äù distortion.
PresetSpeechCosmicInterference	18	Preset that provides a ,ÄúSpeechCosmicInterference,Äù distortion.
PresetSpeechGoldenPi	19	Preset that provides a ,ÄúSpeechGoldenPi,Äù distortion.
PresetSpeechRadioTower	20	Preset that provides a ,ÄúSpeechRadioTower,Äù distortion.
PresetSpeechWaves	21	Preset that provides a ,ÄúSpeechWaves,Äù distortion.

5.23 class AVAudioUnitEffectMBS

5.23.1 class AVAudioUnitEffectMBS

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

Function: An AVAudioUnitEffect class that processes audio in real-time using AudioUnits of type: effect, music effect, panner, remote effect or remote music effect.
Notes: These effects run in real-time and process some number of audio input samples to produce number of audio output samples. A delay unit is an example of an effect unit.
Subclass of the AVAudioUnitMBS class.

5.23.2 Methods

5.23.3 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Initializes an AVAudioUnitEffect object.
Notes: audioComponentDescription: AudioComponentDescription of the AudioUnit to be instantiated.

The audioComponentDescription must be one of these types kAudioUnitType_Effect, kAudioUnitType_MusicEffect, kAudioUnitType_Panner, kAudioUnitType_RemoteEffect, or kAudioUnitType_RemoteMusicEffect.

5.23.4 Properties

5.23.5 bypass as Boolean

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

Function: Bypass state of the AudioUnit.
Notes: (Read and Write property)

5.24 class AVAudioUnitEQFilterParametersMBS

5.24.1 class AVAudioUnitEQFilterParametersMBS

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

Function: The AVAudioUnitEQFilterParameters class encapsulates the parameters used by an AVAudioUnitEQ instance.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

5.24.2 Methods

5.24.3 Constructor

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

Function: The private constructor.

5.24.4 Properties

5.24.5 bandwidth as Double

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

Function: The bandwidth of the EQ filter, in octaves.
Notes: The value range of values is 0.05 to 5.0 octaves.
(Read and Write property)

5.24.6 bypass as Boolean

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

Function: The bypass state of the EQ filter band.
Notes: True if the bypass is active, otherwise false.
(Read and Write property)

5.24.7 filterType as Integer

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

Function: The EQ filter type.
Notes: The default value is AVAudioUnitEQFilterTypeParametric.
(Read and Write property)

5.24.8 frequency as Double

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

Function: The frequency of the EQ filter, in hertz.
Notes: The valid range of values is 20 Hz through (SampleRate/2).
(Read and Write property)

5.24.9 gain as Double

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

Function: The gain of the EQ filter, in decibels.
Notes: The default value is 0 db. The valid range of values is -96 db through 24 db.
(Read and Write property)

5.24.10 Constants

Filter Types

Constant	Value	Description
FilterTypeBandPass	5	Band pass filter. The required parameters are: frequency (center) and bandwidth.
FilterTypeBandStop	6	Band stop filter (‚Äúnotch filter‚Äù). The required parameters are: frequency (center) and bandwidth.
FilterTypeHighPass	2	Simple Butterworth 2nd order high pass filter. The required parameters are: frequency (-3 dB cutoff at specified frequency).
FilterTypeHighShelf	8	High shelf filter. The required parameters are: frequency (center) and gain.
FilterTypeLowPass	1	Simple Butterworth 2nd order low pass filter. The required parameters are: frequency (-3 dB cutoff at specified frequency).
FilterTypeLowShelf	7	Low shelf filter. The required parameters are: frequency (center) and gain.
FilterTypeParametric	0	Parametric filter based on Butterworth analog prototype. The required parameters are: frequency (center), bandwidth, and gain.
FilterTypeResonantHighPass	4	High pass filter with resonance support (using the bandwidth parameter). The required parameters are: frequency (-3 dB cutoff at specified frequency) and bandwidth.
FilterTypeResonantHighShelf	10	High shelf filter with resonance support (using the bandwidth parameter). The required parameters are: frequency (center), bandwidth, and gain.
FilterTypeResonantLowPass	3	Low pass filter with resonance support (using the bandwidth parameter). The required parameters are: frequency (-3 dB cutoff at specified frequency) and bandwidth.
FilterTypeResonantLowShelf	9	Low shelf filter with resonance support (using the bandwidth parameter). The required parameters are: frequency (center), bandwidth, and gain.

5.25 class AVAudioUnitEQMBS

5.25.1 class AVAudioUnitEQMBS

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

Function: The AVAudioUnitEQ class is an AVAudioUnitEffect subclass that implements a multi-band equalizer.
Notes: The AVAudioUnitEQFilterParameters class encapsulates the filter parameters that are returned in the bands property array.
Subclass of the AVAudioUnitEffectMBS class.

5.25.2 Methods

5.25.3 bands as AVAudioUnitEQFilterParametersMBS()

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

Function: An array of AVAudioUnitEQFilterParameters objects.
Notes: The number of elements in the array is equal to the number of bands.

5.25.4 Constructor

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

Function: The constructor.
See also:

- 5.25.5 Constructor(bands as Integer) 850

5.25.5 Constructor(bands as Integer)

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

Function: Initializes a newly allocated AVAudioUnitEQ instance.
Notes: Bands: The number of bands created by the EQ.
See also:

- 5.25.4 Constructor 850

5.25.6 Properties

5.25.7 globalGain as Double

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

Function: The overall gain adjustment applied to the signal, in decibels.
Notes: The default value is 0 db. The valid range of values is -96 db to 24 db.
(Read and Write property)

5.26 class AVAudioUnitGeneratorMBS

5.26.1 class AVAudioUnitGeneratorMBS

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

Function: The AVAudioUnitGenerator is an AVAudioUnit subclass that generates audio output.
Notes: An AVAudioUnitGenerator represents an AudioUnit of type kAudioUnitType_Generator or kAudioUnitType_RemoteGenerator. A generator will have no audio input, but will produce audio output. An example is a tone generator.
Subclass of the AVAudioUnitMBS class.

5.26.2 Methods

5.26.3 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Initializes a newly allocated AVAudioUnitGenerator instance.
Notes: audioComponentDescription: The audio component description.

The AudioComponentDescription struct componentType field must be kAudioUnitType_Generator or kAudioUnitType_RemoteGenerator.

5.26.4 Properties

5.26.5 bypass as Boolean

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

Function: The bypass state of the audio unit.
Notes: True if the bypass state is enabled, otherwise false.
(Read and Write property)

5.26.6 obstruction as Double

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

Function: Simulates filtering of the direct path of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Only the direct path of sound between the source and listener

are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.7 occlusion as Double

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

Function: Simulates filtering of the direct and reverb paths of sound due to an obstacle.
Notes: The value of obstruction is in decibels. Both the direct and reverb paths of sound between the source and listener are blocked.

The default value is 0.0. The range of valid values is -100 to 0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.8 pan as Double

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

Function: The buss's stereo pan.
Notes: The default value is 0.0. A value in the range -1.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.9 position as AVAudio3DPointMBS

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

Function: The location of the source in the 3D environment.
Notes: The coordinates are specified in meters. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.10 rate as Double

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

Function: Changes the playback rate of the input signal.
Notes: A value of 2.0 results in the output audio playing one octave higher. A value of 0.5, results in the output audio playing one octave lower.

The default value is 1.0. The range of valid values is 0.5 to 2.0. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.11 renderingAlgorithm as Integer

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

Function: Type of rendering algorithm used.
Notes: Depending on the current output format of the AVAudioEnvironmentNode instance, only a subset of the rendering algorithms may be supported. An array of valid rendering algorithms is retrieved by calling the applicableRenderingAlgorithms function of the AVAudioEnvironmentNode instance.

The default rendering algorithm is AVAudio3DMixingRenderingAlgorithmEqualPowerPanning. This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.12 reverbBlend as Double

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

Function: Controls the blend of dry and reverb processed audio.
Notes: This property controls the amount of the source,Äôs audio that will be processed by the reverb by the AVAudioEnvironmentNode instance. A value of 0.5 will result in an equal blend of dry and processed (wet) audio.

The default is 0.0. The range of valid values is 0.0 (completely dry) to 1.0 (completely wet). This property is currently implemented only by the AVAudioEnvironmentNode class mixer.
(Read and Write property)

5.26.13 volume as Double

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

Function: The buss's input volume.
Notes: The default value is 1.0. The range of valid values are 0.0 to 1.0. This property is currently implemented only by the AVAudioEnvironmentNode and AVAudioMixerNode class mixers.

(Read and Write property)

5.27 class AVAudioUnitMBS

5.27.1 class AVAudioUnitMBS

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

Function: The AVAudioUnit class is a subclass of the AVAudioNode class that, depending on the type of the audio unit, processes audio either in real-time or non real-time.
Notes: Subclass of the AVAudioNodeMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.3pr3

5.27.2 Methods

5.27.3 AddPropertyListener(ID as UInt32)

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

Function: Registeres to receive notifications for when a property changes.
Notes: When an audio unit property value changes, a PropertyListener event can be called by the audio unit to inform interested parties that this event has occurred.

You must call RemovePropertyListener when you are done with the audio unit.

Lasterror is set.

5.27.4 Constructor

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

Function: The private constructor.
See also:

- 5.27.5 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS) 856

5.27.5 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Private constructor. Initializes a newly allocated audio component specified by the description.
Example:

```
// Needs OS X 10.10 or newer
dim m as new AVAudioUnitComponentManagerMBS

dim a() as AVAudioUnitComponentMBS = m.allComponents

for each c as AVAudioUnitComponentMBS in a
dim d as AVAudioComponentDescriptionMBS = c.audioComponentDescription
'List.AddRow c.Name, c.ManufacturerName, c.LocalizedTypeName, c.VersionString, d.componentType+"
"+d.componentSubType

// we look for the mixer
// AUMixer Apple Mixer 1.6.0 aumx smxr

if d.componentType = "aumx" and d.componentSubType = "smxr" then
dim aa as new AVAudioUnitMBS(d)
MsgBox aa.Name
end if

next
```

Notes: audioComponentDescription: The description of the audio unit to be initialized.
See also:

- 5.27.4 Constructor 856

5.27.6 CreateView(PreferredSize as NSSizeMBS) as NSViewMBS

Plugin Version: 18.2, Platform: macOS, Targets: Desktop only.

Function: Creates view if possible for the audio unit.
Example:

```
// audiounit is property of window
dim audiounit as new AVAudioUnitTimePitchMBS

dim preferredSize as new NSSizeMBS(200,100)
// view is property of window
dim view as NSViewMBS = audiounit.CreateView(preferredSize)

dim w as new NSWindowMBS(self)

// set position
view.setFrameOrigin(new NSPointMBS(100, 100))

// add to window
w.contentView.addSubview view
```

Notes: Returns nil in case of error.
PreferredSize is the preferred size of the control.

5.27.7 Destructor

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

Function: The destructor.

5.27.8 GetParameter(ID as UInt32, Scope as UInt32, Element as UInt32) as Single

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

Function: Get the value of a parameter as specified by its ID, scope and element.
Notes: Lasterror is set.
If no error occurs, the result is the parameter value.
Please check Apples documentation for possible parameters.

5.27.9 GetProperty(ID as UInt32, Scope as UInt32, Element as UInt32) as Memoryblock

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

Function: Retrieves the value of a specified property.
Notes: The plugin queries the data size, creates a memoryblock, queries the value and returns it.
Lasterror is set.

5.27.10 GetPropertyInfo(ID as UInt32, Scope as UInt32, Element as UInt32, byref WriteAble as Boolean) as UInt32

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

Function: Retrieves information about a specified property.
Notes: The API can be used to retrieve both the size of the property, and whether it is writable or not. In order to get a general answer on the capability of an audio unit, this function should be called before the audio unit is initialized (as some properties are writable when the audio unit is initialized, and others not)

Lasterror is set.

5.27.11 installLevelMonitor(CallsPerSecond as Integer, tag as Variant = nil)

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

Function: Installs the handler to call LevelMonitor event regularly.
Notes: CallsPerSecond can be between 5 and 50. Tag is stored and passed to the event.

5.27.12 RemovePropertyListener(ID as UInt32)

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

Function: Remove a previously registered property listener.

5.27.13 SetParameter(ID as UInt32, Scope as UInt32, Element as UInt32, Value as Single, inBufferOffsetInFrames as UInt32 = 0)

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

Function: Set the value of a parameter as specified by its ID, scope and element.
Notes: Parameter IDs are consistent across all of the elements in a scope - so for a mixer, the "input volume" parameter can be applied on any input, and the particular input is specified by the elementID.

Lasterror is set.
Please check Apples documentation for possible parameters.

5.27.14 SetProperty(ID as UInt32, Scope as UInt32, Element as UInt32, data as Memoryblock)

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

Function: Sets the value of a specified property.
Notes: Some properties can be cleared by passing data = nil.
Lasterror is set.

5.27.15 Properties

5.27.16 audioComponentDescription as AVAudioComponentDescriptionMBS

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

Function: The audio compression description of the underlying Core Audio audio unit.
Notes: (Read only property)

5.27.17 audioUnitHandle as Integer

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

Function: The internal handle for the audio unit.
Notes: (Read only property)

5.27.18 lastError as Integer

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

Function: The last error code for the property/parameter functions.
Notes: Zero means no error.
(Read and Write property)

5.27.19 ManufacturerName as String

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

Function: The name of the manufacturer of the audio unit.
Notes: (Read only property)

5.27.20 Name as String

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

Function: The name of the audio unit.
Notes: (Read only property)

5.27.21 Version as Integer

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

Function: The version number of the audio unit.
Notes: (Read only property)

5.27.22 Events

5.27.23 LevelMonitor(Level0 as Double, Level1 as Double, Level2 as Double, Level3 as Double, Level4 as Double, Level5 as Double, Level6 as Double, Level7 as Double, tag as Variant)

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

Function: Reports level values.
Notes: We pass 8 level parameters for up to 8 channels.

5.27.24 PropertyListener(ID as UInt32, Scope as UInt32, Element as UInt32)

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

Function: The property listener event.
Notes: When you listen for changes on a property, this event can be called.
Either synchronous on main thread or when received on other threads, the plugin will run it asynchronously on main thread.

5.28 class AVAudioUnitMIDIInstrumentMBS

5.28.1 class AVAudioUnitMIDIInstrumentMBS

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

Function: The AVAudioUnitMIDIInstrument class is an abstract class representing music devices or remote instruments.
Notes: An AVAudioUnitMIDIInstrument subclass can be used in a chain that processes realtime input (live) and has general concept of music events, for example, notes.

Available in OS X v10.10 and later.
Subclass of the AVAudioUnitMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.4pr1

5.28.2 Methods

5.28.3 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Initializes a MIDI instrument audio unit with the specified component description.
Notes: description: The AudioComponentDescription struct. The component of type must be kAudioUnitType_MusicDevice or kAudioUnitType_RemoteInstrument.

5.28.4 sendController(Controller as Integer, Value as Integer, Channel as Integer)

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

Function: Send a MIDI ‚Äúcontroller‚Äù event to the instrument.
Notes: controller: Specifies a standard MIDI controller number. The valid range is 0 to 127.
value: Value for the controller. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

5.28.5 sendMIDIEvent(midiStatus as Integer, data1 as Integer)

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

Function: Sends a MIDI event which contains one data byte to the instrument.
Notes: midiStatus: The STATUS value of the MIDI event.
data1: The data byte of the MIDI event.
See also:

- 5.28.6 sendMIDIEvent(midiStatus as Integer, data1 as Integer, data2 as Integer) 863

5.28.6 sendMIDIEvent(midiStatus as Integer, data1 as Integer, data2 as Integer)

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

Function: Sends a MIDI event which contains two data bytes to the instrument.
Notes: midiStatus: The STATUS value of the MIDI event.
data1: The first data byte of the MIDI event.
data2: The first data byte of the MIDI event.
See also:

- 5.28.5 sendMIDIEvent(midiStatus as Integer, data1 as Integer) 862

5.28.7 sendMIDISysExEvent(data as MemoryBlock)

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

Function: Sends a MIDI ,ÄúSystem Exclusive,Äù event to the instrument.
Notes: midiData: A memoryblock object containing the complete SysEx data including start(F0) and termination(F7) bytes.

5.28.8 sendPitchBend(pitchbend as Integer, Channel as Integer)

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

Function: Sends MIDI ,ÄúPitch Bend,Äù event to the instrument.
Notes: pitchbend: Value of the pitchbend. The valid range of values is 0 to 16383.
channel: The channel number to which the event is sent.

If this method is not invoked, then the pitch bend is the default value of 8192 (no pitch).

5.28.9 sendPressure(pressure as Integer, Channel as Integer)

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

Function: Sends MIDI ‚Äúchannel pressure‚Äù event to the instrument.
Notes: pressure: The value of the pressure. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

5.28.10 sendPressureForKey(Key as Integer, value as Integer, Channel as Integer)

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

Function: Sends MIDI ‚ÄúPolyphonic key pressure‚Äù event to the instrument.
Notes: key: The key (note) number to which the pressure event applies. The valid range is 0 to 127.
value: The value of the pressure. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

5.28.11 sendProgramChange(program as Integer, bankMSB as Integer, bankLSB as Integer, Channel as Integer)

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

Function: Sends a MIDI ‚ÄúProgram Change‚Äù and ‚ÄúBank Select‚Äù events to the instrument.
Notes: program: Specifies the program (preset) number within the bank to load. The valid range is 0 to 127.
bankMSB: Specifies the most significant byte value for the bank to select. The valid range is 0 to 127.
bankLSB: Specifies the least significant byte value for the bank to select. The valid range is 0 to 127.
channel: The channel number to which the event is sent.
See also:

- 5.28.12 sendProgramChange(program as Integer, Channel as Integer) 864

5.28.12 sendProgramChange(program as Integer, Channel as Integer)

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

Function: Sends a MIDI ‚ÄúProgram Change‚Äù and ‚ÄúBank Select‚Äù events to the instrument.
Notes: program: Specifies the program. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

The instrument will be loaded from the bank that has been previous set by MIDI ‚ÄúBank Select‚Äù controller messages (0 and 31). If none has been set, bank 0 will be used.
See also:

- 5.28.11 sendProgramChange(program as Integer, bankMSB as Integer, bankLSB as Integer, Channel as Integer) 864

5.28.13 startNote(note as Integer, Velocity as Integer, Channel as Integer)

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

Function: Sends a MIDI ,ÄúNote On,Äù event to the instrument.
Notes: note: The note number (key) to play. The valid range is 0 to 127.
velocity: Specifies the volume at which the note is played. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

5.28.14 stopNote(note as Integer, Channel as Integer)

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

Function: Sends a MIDI ,ÄúNote Off,Äù event to the instrument
Notes: note: The note number (key) to stop. The valid range is 0 to 127.
channel: The channel number to which the event is sent.

5.29 class AVAudioUnitReverbMBS

5.29.1 class AVAudioUnitReverbMBS

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

Function: The AVAudioUnitReverb class is an AVAudioUnitEffect subclass that implements a reverb effect.
Notes: A reverb simulates the acoustic characteristics of a particular environment. Use the different presets to simulate a particular space and blend it in with the original signal using the wetDryMix property.
Subclass of the AVAudioUnitEffectMBS class.

5.29.2 Methods

5.29.3 Constructor

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

Function: The constructor.

5.29.4 loadFactoryPreset(preset as Integer)

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

Function: Configures the audio unit a reverb preset.
Notes: preset: The reverb preset. See AVAudioUnitReverbPreset for the supported values.

The default value is AVAudioUnitReverbPresetMediumHall.

5.29.5 Properties

5.29.6 wetDryMix as Double

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

Function: The blend of the wet and dry signals.
Notes: The blend is specified as a percentage. The range is 0% (all dry) through 100% (all wet).
(Read and Write property)

5.29.7 Constants

Reverb presets

Constant	Value	Description
PresetCathedral	8	The reverb preset with the acoustic characteristics of a cathedral environment.
PresetLargeChamber	7	The reverb preset with the acoustic characteristics of a large-sized chamber environment.
PresetLargeHall	4	The reverb preset with the acoustic characteristics of a large-sized hall environment.
PresetLargeHall2	12	The reverb preset with the acoustic characteristics of an alternate large-sized hall environment.
PresetLargeRoom	2	The reverb preset with the acoustic characteristics of a large-sized room environment.
PresetLargeRoom2	9	The reverb preset with the acoustic characteristics of an alternate large-sized room environment.
PresetMediumChamber	6	The reverb preset with the acoustic characteristics of a medium-sized chamber environment.
PresetMediumHall	3	The reverb preset with the acoustic characteristics of a medium-sized hall environment. This is the default value for the audio unit.
PresetMediumHall2	10	The reverb preset with the acoustic characteristics of an alternate medium-sized hall environment.
PresetMediumHall3	11	The reverb preset with the acoustic characteristics of an alternate medium-sized hall environment.
PresetMediumRoom	1	The reverb preset with the acoustic characteristics of a medium-sized room environment.
PresetPlate	5	The reverb preset with the acoustic characteristics of a plate environment.
PresetSmallRoom	0	The reverb preset with the acoustic characteristics of a small-sized room environment.

5.30 class AVAudioUnitSamplerMBS

5.30.1 class AVAudioUnitSamplerMBS

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

Function: The AVAudioUnitSampler class encapsulates Apple's Sampler Audio Unit.
Notes: The sampler audio unit can be configured by loading different types of instruments such as an ‚Äúaupreset‚Äù file, a DLS or SF2 sound bank, an EXS24 instrument, a single audio file or with an array of audio files. The output is a single stereo bus.
Subclass of the AVAudioUnitMIDIInstrumentMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.4pr1

5.30.2 Methods

5.30.3 Constructor

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

Function: The private constructor.
See also:

- 5.30.4 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS) 868

5.30.4 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Initializes a MIDI instrument audio unit with the specified component description.
See also:

- 5.30.3 Constructor 868

5.30.5 loadAudioFilesAtFiles(Files() as folderitem, byref error as NSErrorMBS) as boolean

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

Function: Configures the sampler by loading the specified audio files.
Notes: Files: An array of files for the audio files.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true if the sampler was configured with the audio file(s); otherwise false.

The audio files are loaded into a new instrument with each audio file placed into its own sampler zone. Any information contained in the audio file regarding their placement in the instrument, for example, root key, key range, will be used.

5.30.6 loadAudioFilesAtURLs(URLs() as string, byref error as NSErrorMBS) as boolean

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

Function: Configures the sampler by loading the specified audio files.
Notes: URLs: An array of URLs for the audio files.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true if the sampler was configured with the audio file(s); otherwise false.

The audio files are loaded into a new instrument with each audio file placed into its own sampler zone. Any information contained in the audio file regarding their placement in the instrument, for example, root key, key range, will be used.

5.30.7 loadInstrumentAtFile(File as folderitem, byref error as NSErrorMBS) as boolean

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

Function: Configures the sampler with the specified instrument file.
Notes: File: The file containing the instrument.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true if the sampler was configured with the instrument file; otherwise false.

The instrument can be of one of the following types: Logic or GarageBand EXS24, sampler‚Äôs native .aupreset file or an audio file, for example, ‚Äú.caf‚Äù, ‚Äú.aiff‚Äù, ‚Äú.wav‚Äù, or ‚Äú.mp3‚Äù.

In the case of single audio file, it is loaded into a new default instrument and any information contained in the audio file regarding its placement in the instrument, for example, root key, key range, will be used.

5.30.8 loadInstrumentAtURL(URL as string, byref error as NSErrorMBS) as boolean

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

Function: Configures the sampler with the specified instrument file.
Notes: URL: The URL of the file containing the instrument.
Error: Returns, by-reference, a description of the error, if an error occurs.

Returns true if the sampler was configured with the instrument file; otherwise false.

The instrument can be of one of the following types: Logic or GarageBand EXS24, sampler,Äôs native .aupreset file or an audio file, for example, ,Äú.caf,Äù, ,Äú.aiff,Äù, ,Äú.wav,Äù, or ,Äú.mp3,Äù.

In the case of single audio file, it is loaded into a new default instrument and any information contained in the audio file regarding its placement in the instrument, for example, root key, key range, will be used.

5.30.9 loadSoundBankInstrumentAtFile(bankFile as folderitem, program as Integer, bankMSB as Integer, bankLSB as Integer, byref error as NSErrorMBS) as boolean

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

Function: loads a specific instrument from the specified sound bank.
Notes: bankFile: Soundbank file. The file can be either a DLS bank (.dls) or a SoundFont bank (.sf2).
program: program number for the instrument to load
bankMSB: MSB for the bank number for the instrument to load. This is usually 0x79 for melodic instruments and 0x78 for percussion instruments.
bankLSB: LSB for the bank number for the instrument to load. This is often 0, and represents the "bank variation".
Error: the status of the operation

This method reads from file and allocates memory, so it should not be called on a real time thread.

5.30.10 loadSoundBankInstrumentAtURL(bankURL as string, program as Integer, bankMSB as Integer, bankLSB as Integer, byref error as NSErrorMBS) as boolean

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

Function: loads a specific instrument from the specified sound bank.
Notes: bankURL: URL for a Soundbank file. The file can be either a DLS bank (.dls) or a SoundFont bank

(.sf2).
program: program number for the instrument to load
bankMSB: MSB for the bank number for the instrument to load. This is usually 0x79 for melodic instruments and 0x78 for percussion instruments.
bankLSB: LSB for the bank number for the instrument to load. This is often 0, and represents the "bank variation".
Error: the status of the operation

This method reads from file and allocates memory, so it should not be called on a real time thread.

5.30.11 Properties

5.30.12 globalTuning as Double

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

Function: Adjusts the tuning of all the notes played.
Notes: The tuning unit is cents. The default value is 0.0. The range of valid values is -2400 to 2400 cents.
(Read and Write property)

5.30.13 masterGain as Double

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

Function: Adjusts the gain, in decibels, of all the notes played.
Notes: The default value is 0.0 db. The range of valid values is -90.0 db to 12.0 db.
(Read and Write property)

5.30.14 stereoPan as Double

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

Function: Adjusts the stereo panning for all the notes played.
Notes: The default value is 0.0. The range of valid values is -1.0 to 1.0.
(Read and Write property)

5.31 class AVAudioUnitTimeEffectMBS

5.31.1 class AVAudioUnitTimeEffectMBS

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

Function: The AVAudioUnitTimeEffect class is an AVAudioUnit subclass that processes audio in non-realtime.
Notes: An AVAudioUnitTimeEffect instance represents an AVAudioUnit of type ‚Äúaufc‚Äù (kAudioUnitType_FormatConverter). These effects do not process audio in real-time. The AVAudioUnitVarispeed class is an example of a time effect unit.
Subclass of the AVAudioUnitMBS class.

5.31.2 Methods

5.31.3 Constructor(audioComponentDescription as AVAudioComponentDescriptionMBS)

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

Function: Initializes a newly allocated audio component specified by the description.
Notes: audioComponentDescription: The description of the audio unit to be initialized. The componentType field of the description struct must be kAudioUnitType_FormatConverter (‚Äúaufc‚Äù).

If the componentType field of the audioComponentDescription struct is not kAudioUnitType_FormatConverter an exception is raised.

5.31.4 Properties

5.31.5 bypass as Boolean

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

Function: Returns the bypass state of the audio unit.
Notes: If true then the audio unit processing is being bypassed.
(Read and Write property)

5.32 class AVAudioUnitTimePitchMBS

5.32.1 class AVAudioUnitTimePitchMBS

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

Function: The AVAudioUnitTimePitch class is an AVAudioUnitTimeEffect subclass that provides good quality playback rate and pitch shifting independent of each other.
Notes: Subclass of the AVAudioUnitTimeEffectMBS class.

5.32.2 Methods

5.32.3 Constructor

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

Function: The constructor.

5.32.4 Properties

5.32.5 overlap as Double

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

Function: The amount of overlap between segments of the input audio signal.
Notes: A higher value results in fewer artifacts in the output signal.

The default value is 8.0. The range of values is 3.0 to 32.0.
(Read and Write property)

5.32.6 pitch as Double

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

Function: The amount by which the input signal is pitch shifted.
Notes: The pitch is measured in ‚Äúcents‚Äù, a logarithmic value used for measuring musical intervals. One octave is equal to 1200 cents. One musical semitone is equal to 100 cents.

The default value is 1.0. The range of values is -2400 to 2400.
(Read and Write property)

5.32.7 rate as Double

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

Function: The playback rate of the input signal.
Notes: The default value is 1.0. The range of supported values is 1/32 to 32.0.
(Read and Write property)

5.33 class AVAudioUnitVarispeedMBS

5.33.1 class AVAudioUnitVarispeedMBS

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

Function: The AVAudioUnitVarispeed class is an AVAudioUnitTimeEffect subclass that allows control of the playback rate.
Notes: Subclass of the AVAudioUnitTimeEffectMBS class.

5.33.2 Methods

5.33.3 Constructor

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

Function: The constructor.

5.33.4 Properties

5.33.5 Rate as Double

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

Function: The audio playback rate.
Notes: The varispeed audio unit resamples the input signal, as a result changing the playback rate also changes the pitch. For example, changing the rate to 2.0 results in the output audio playing one octave higher. Similarly changing the rate to 0.5, results in the output audio playing one octave lower.

The pitch is measured in ,Äúcents,Äù, a logarithmic value used for measuring musical intervals. One octave is equal to 1200 cents. One musical semitone is equal to 100 cents.

Using the rate value you calculate the pitch (in cents) using the formula pitch = 1200.0 * log2(rate). Conversely, you calculate the appropriate rate for a desired pitch with the formula rate = pow(2, cents/1200.0).

The default value is 1.0. The range of values is 0.25 to 4.0. The unit of rate is unspecified.
(Read and Write property)

Chapter 6

AVMovie

6.1 class AVFragmentedMovieMBS

6.1.1 class AVFragmentedMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The movie class for fragmented movies.
Notes: A subclass of AVMovie for handling fragmented movie files. An AVFragmentedMovie is capable of changing the values of certain of its properties and those of its tracks, if it's associated with an instance of AVFragmentedMovieMinder when one or more movie fragments are appended to the movie file.
Subclass of the AVMovieMBS 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 / Real Studio Plugins, version 16.0pr5

6.1.2 Methods

6.1.3 AVFragmentedMovieContainsMovieFragmentsDidChangeNotification as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names.
Notes: Posted after the value of @"containsMovieFragments" has already been loaded and the AVFragmentedMovie is added to an AVFragmentedMovieMinder, either when 1) movie fragments are detected in the movie file on disk after it had previously contained none or when 2) no movie fragments are detected in the movie file on disk after it had previously contained one or more.

6.1.4 AVFragmentedMovieDurationDidChangeNotification as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names.
Notes: Posted when the duration of an AVFragmentedMovie changes while it's being minded by an AVFragmentedMovieMinder, but only for changes that occur after the status of the value of @"duration" has reached AVKeyValueStatusLoaded.

6.1.5 AVFragmentedMovieWasDefragmentedNotification as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names.
Notes: Posted when the movie file on disk is defragmented while an AVFragmentedMovie is being minded by an AVFragmentedMovieMinder, but only if the defragmentation occurs after the status of the value of @"canContainMovieFragments" has reached AVKeyValueStatusLoaded.

6.1.6 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.

6.1.7 fragmentedMovieTracks as AVFragmentedMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The tracks in a movie.
Notes: The value of this property is an array of tracks the movie contains; the tracks are of type AVFragmentedMovieTrack.

6.1.8 fragmentedMovieTracksWithMediaCharacteristic(mediaCharacteristic as string) as AVFragmentedMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVFragmentedMovieTracks of the asset that present media with the specified characteristic.
Notes: mediaCharacteristic: The media characteristic according to which the receiver filters its AVFragmentedMovieTracks. (Media characteristics are defined in AVMediaFormat.h)

An array of AVFragmentedMovieTracks; may be empty if no tracks with the specified characteristic are available.
Becomes callable without blocking when the key ”tracks” has been loaded

6.1.9 fragmentedMovieTracksWithMediaType(mediaType as string) as AVFragmentedMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVFragmentedMovieTracks of the asset that present media of the specified media type.
Notes: mediaType: The media type according to which the receiver filters its AVFragmentedMovieTracks. (Media types are defined in AVMediaFormat.h)
An array of AVFragmentedMovieTracks; may be empty if no tracks of the specified media type are available.
Becomes callable without blocking when the key ”tracks” has been loaded

6.1.10 fragmentedMovieTrackWithTrackID(ID as Integer) as AVFragmentedMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an instance of AVFragmentedMovieTrack that represents the track of the specified trackID.
Notes: ID: The trackID of the requested AVFragmentedMovieTrack.

An instance of AVFragmentedMovieTrack; may be nil if no track of the specified trackID is available.
Becomes callable without blocking when the key ”tracks” has been loaded

6.2 class AVFragmentedMovieTrackMBS

6.2.1 class AVFragmentedMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The class for a movie track from a fragmented movie.
Notes: A subclass of AVMovieTrack for handling tracks of fragmented movie files. An AVFragmentedMovieTrack is capable of changing the values of certain of its properties, if its parent movie is associated with an instance of AVFragmentedMovieMinder when one or more movie fragments are appended to the movie file.
Subclass of the AVMovieTrackMBS 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 / Real Studio Plugins, version 16.0pr5

6.2.2 Methods

6.2.3 AVFragmentedMovieTrackSegmentsDidChangeNotification as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names for fragmented movies.
Notes: Posted when the array of segments of an AVFragmentedMovieTrack changes while the associated instance of AVFragmentedMovie is being minded by an AVFragmentedMovieMinder, but only for changes that occur after the status of the value of @"segments" has reached AVKeyValueStatusLoaded.

6.2.4 AVFragmentedMovieTrackTimeRangeDidChangeNotification as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names for fragmented movies.
Notes: Posted when the timeRange of an AVFragmentedMovieTrack changes while the associated instance of AVFragmentedMovie is being minded by an AVFragmentedMovieMinder, but only for changes that occur after the status of the value of "timeRange" has reached AVKeyValueStatusLoaded.

6.2.5 AVFragmentedMovieTrackTotalSampleDataLengthDidChangeNotification as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the notification names for fragmented movies.
Notes: This notification name has been deprecated. Use either AVFragmentedMovieTrackTimeRange-

DidChangeNotification or AVFragmentedMovieTrackSegmentsDidChangeNotification instead; in either case, you can assume that timing changes to fragmented tracks result in changes to the total length of the sample data used by the track.

6.2.6 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.

6.3 class AVMediaDataStorageMBS

6.3.1 class AVMediaDataStorageMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The class for media storage.
Notes: Available on OS X 10.11 and newer.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.0pr5

6.3.2 Methods

6.3.3 available as boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Example:

```
if not AVMediaDataStorageMBS.available then
MsgBox "This application requires a newer OS X version."
end if
```

Notes: Returns true on OS X 10.11.

6.3.4 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.3.5 Constructor(File as FolderItem, Options as Dictionary = nil) 882
- 6.3.6 Constructor(URL as String, Options as Dictionary = nil) 883

6.3.5 Constructor(File as FolderItem, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMediaDataStorage object associated with a folderitem.
Notes: File: A folderitem that specifies a file where sample data that is added to a movie or track should

be written.
options: A dictionary object that contains keys for specifying options for the initialization of the AVMediaDataStorage object. Currently no keys are defined.

Raises exception if things go wrong.
See also:

- 6.3.4 Constructor 882
- 6.3.6 Constructor(URL as String, Options as Dictionary = nil) 883

6.3.6 Constructor(URL as String, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMediaDataStorage object associated with a file URL.
Notes: URL: An URL that specifies a file where sample data that is added to a movie or track should be written.
options: A dictionary object that contains keys for specifying options for the initialization of the AVMediaDataStorage object. Currently no keys are defined.

Raises exception if things go wrong.
See also:

- 6.3.4 Constructor 882
- 6.3.5 Constructor(File as FolderItem, Options as Dictionary = nil) 882

6.3.7 Properties

6.3.8 Handle as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

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

6.3.9 URL as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The URL from which the receiver was initialized; may be "".
Notes: (Read only property)

6.4 class AVMovieMBS

6.4.1 class AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: AVMovie represents the audiovisual containers in a file that conforms to the QuickTime movie file format or to one of the related ISO base media file formats (such as MPEG-4).
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

MsgBox str(m.duration.Seconds)+" seconds."
```

Notes: AVMovie supports operations involving the format-specific portions of the QuickTime Movie model that are not supported by AVAsset. For instance, you can retrieve the movie header from an existing QuickTime movie file. You can also use AVMovie to write a movie header into a new file, thereby creating a reference movie.

A mutable subclass of AVMovie, AVMutableMovie, provides methods that support the familiar Movie editing model; for instance, you can use AVMutableMovie to copy media data from one track and paste that data into another track. You can also use AVMutableMovie to establish track references from one track to another (for example, to set one track as a chapter track of another track). If you want to perform editing operations on individual tracks, you can use the associated classes AVMovieTrack and AVMutableMovieTrack.

You need to use AVMovie and AVMutableMovie only when operating on format-specific features of a QuickTime or ISO base media file. You generally do not need to use these classes just to open and play QuickTime movie files or ISO base media files. Instead, you can use the classes AVURLAsset and AVPlayerItem. If however you already have an AVMutableMovie and want to play it or inspect it, you can make an immutable snapshot of the AVMutableMovie.
Subclass of the AVAssetMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.0pr5

6.4.2 Methods

6.4.3 available as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Example:

```
if not AVMovieMBS.available then
MsgBox "This application requires a newer OS X version."
end if
```

Notes: Returns true on OS X 10.10.

6.4.4 AVMovieReferenceRestrictionsKey as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Indicates the restrictions used by the movie when resolving references to external media data.
Notes: The value of this key is an NSNumber wrapping an AVAssetReferenceRestrictions enum value or the logical combination of multiple such values. See AVAsset.h for the declaration of the AVAssetReferenceRestrictions enum.
Some movies can contain references to media data stored outside the movie's container, for example in another file. This key can be used to specify a policy to use when these references are encountered. If a movie contains one or more references of a type that is forbidden by the reference restrictions, loading of movie properties will fail. In addition, such a movie cannot be used with other AVFoundation modules, such as AVPlayerItem or AVAssetExportSession.

6.4.5 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.
See also:

- 6.4.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 885
- 6.4.7 Constructor(File as FolderItem, Options as Dictionary = nil) 886
- 6.4.8 Constructor(URL as String, Options as Dictionary = nil) 887

6.4.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in an data parameter.
Notes: data: An Memoryblock containing a movie header. We make a copy of that.
options: Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.

You can use this method to operate on movie headers that are not stored in files; this might include movie headers on the pasteboard (which do not contain media data). In general you should avoid loading an entire movie file with its media data into an instance of Memoryblock/String! By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

In case of error raises an exception.

Available on OS X 10.11.
See also:

- 6.4.5 Constructor 885
- 6.4.7 Constructor(File as FolderItem, Options as Dictionary = nil) 886
- 6.4.8 Constructor(URL as String, Options as Dictionary = nil) 887

6.4.7 Constructor(File as FolderItem, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.m4v”)
dim m as new AVMovieMBS(f, nil)

MsgBox str(m.duration.Seconds)+” seconds.”
```

Notes: File: A folderitem that specifies a file containing a movie header.
options: A Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.

By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

Raises exception if something goes wrong.
See also:

- 6.4.5 Constructor 885

- 6.4.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 885
- 6.4.8 Constructor(URL as String, Options as Dictionary = nil) 887

6.4.8 Constructor(URL as String, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: URL: An URL that specifies a file containing a movie header.
options: A Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

Raises exception if something goes wrong.
See also:

- 6.4.5 Constructor 885
- 6.4.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 885
- 6.4.7 Constructor(File as FolderItem, Options as Dictionary = nil) 886

6.4.9 copy as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates a copy of the movie.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

dim c as AVMutableMovieMBS = m.mutableCopy

Break // now edit

dim k as AVMovieMBS = c.copy

break // now we have k with snapshot and can still edit c
```

6.4.10 movieHeaderWithFileType(fileType as String, byref error as NSErrorMBS) as MemoryBlock

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates a memoryblock object containing the movie header of the AVMovie object.
Notes: fileType: A UTI indicating the specific file format of the movie header (e.g. AVFileTypeQuickTimeMovie for a QuickTime movie).
Error: If an error occurs reading the movie header, describes the nature of the failure.

The movie header will be a pure reference movie, with no base URL, suitable for use on the pasteboard.
Returns memoryblock in case of success or nil in case of error.

Available on OS X 10.11 or newer.

6.4.11 movieTracks as AVMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The tracks in a movie.
Notes: The value of this property is an array of tracks the movie contains; the tracks are of type AVMovieTrack.

6.4.12 movieTracksWithMediaCharacteristic(mediaCharacteristic as string) as AVMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVMovieTracks of the asset that present media with the specified characteristic.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

dim videotracks() as AVMovieTrackMBS = m.movieTracksWithMediaCharacteristic(AVFoundationMBS.AVMediaCharacteristicVisual)

MsgBox str(UBound(videotracks)+1)+" visual tracks"
```

Notes: mediaCharacteristic: The media characteristic according to which the receiver filters its AVMovieTracks. (Media characteristics are defined in AVMediaFormat.h)

An array of AVMovieTracks; may be empty if no tracks with the specified characteristic are available.
Becomes callable without blocking when the key ”tracks” has been loaded.

6.4.13 movieTracksWithMediaType(mediaType as string) as AVMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVMovieTracks of the asset that present media of the specified media type.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.m4v”)
dim m as new AVMovieMBS(f, nil)

dim videotracks() as AVMovieTrackMBS = m.movieTracksWithMediaType(AVFoundationMBS.AVMediaTypeVideo)

MsgBox str(UBound(videotracks)+1)+” video tracks”
```

Notes: mediaType: The media type according to which the receiver filters its AVMovieTracks. (Media types are defined in AVMediaFormat.h)
An array of AVMovieTracks; may be empty if no tracks of the specified media type are available.
Becomes callable without blocking when the key ”tracks” has been loaded.

6.4.14 movieTrackWithTrackID(ID as Integer) as AVMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an instance of AVMovieTrack that represents the track of the specified trackID.
Notes: ID: The trackID of the requested AVMovieTrack.

Returns n instance of AVMovieTrack; may be nil if no track of the specified trackID is available.
Becomes callable without blocking when the key ”tracks” has been loaded

6.4.15 movieTypes as String()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides the file types the AVMovie class understands.
Example:

```
MsgBox join(AVMovieMBS.movieTypes, EndOfLine)
```

Notes: Returns an array of UTIs identifying the file types the AVMovie class understands.

6.4.16 movieWithData(Data as MemoryBlock, Options as Dictionary = nil) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in an data parameter.
Notes: data: An Memoryblock/String containing a movie header. We make a copy of that.
options: Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
You can use this method to operate on movie headers that are not stored in files; this might include movie headers on the pasteboard (which do not contain media data). In general you should avoid loading an entire movie file with its media data into an instance of Memoryblock/String! By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.4.17 movieWithData(Data as String, Options as Dictionary = nil) as AVMovieMBS 890

6.4.17 movieWithData(Data as String, Options as Dictionary = nil) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in an data parameter.
Example:

```
// load movie in memory
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim b as BinaryStream = BinaryStream.Open(f)

// now open it from memory
dim s as string = b.Read(b.Length)
dim m as AVMovieMBS = AVMovieMBS.movieWithData(s)

MsgBox str(m.duration.Seconds)+" seconds."
```

Notes: data: An Memoryblock/String containing a movie header. We make a copy of that.
options: Dictionary object that contains keys for specifying options for the initialization of the AVMovie

object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
You can use this method to operate on movie headers that are not stored in files; this might include movie headers on the pasteboard (which do not contain media data). In general you should avoid loading an entire movie file with its media data into an instance of Memoryblock/String! By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.4.16 movieWithData(Data as MemoryBlock, Options as Dictionary = nil) as AVMovieMBS 890

6.4.18 movieWithFile(File as FolderItem, Options as Dictionary = nil) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as AVMovieMBS = AVMovieMBS.movieWithFile(f)

MsgBox str(m.duration.Seconds)+" seconds."
```

Notes: File: A folderitem that specifies a file containing a movie header.
options: A Dictionary object that contains keys for specifying options for the initialization of the AVMovie object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

6.4.19 movieWithURL(URL as String, Options as Dictionary = nil) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: URL: An URL that specifies a file containing a movie header.
options: A Dictionary object that contains keys for specifying options for the initialization of the AVMovie

object. Currently no keys are defined.
Returns An AVMovie object or nil in case of error.
By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

6.4.20 mutableCopy as AVMutableMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates a mutable copy of the movie.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

dim c as AVMutableMovieMBS = m.mutableCopy

Break // now edit

dim k as AVMovieMBS = c.copy

break // now we have k with snapshot and can still edit c
```

6.4.21 writeMovieHeaderToFile(File as FolderItem, fileType as String, options as Integer, byref error as NSErrorMBS) as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Writes the movie header to a destination file.
Notes: File: A folderitem indicating where to write the movie header.
fileType: A UTI indicating the specific file format (e.g. AVFileTypeQuickTimeMovie for a QuickTime movie).
options: An integer whose bits specify options for the writing of the movie header. Can be AVMovieWritingTruncateDestinationToMovieHeaderOnly.
Error: If an error occurs writing the movie header, describes the nature of the failure.

Data references in the output movie header are adjusted to be relative to the destination URL. Note that modifications to instances of AVMutableMovie, to their constituent AVMutableMovieTracks, or to their collections of metadata are committed to storage when their movie headers are written.

6.4.22 writeMovieHeaderToURL(URL as String, fileType as String, options as Integer, byref error as NSErrorMBS) as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Writes the movie header to a destination URL.
Notes: URL: An URL indicating where to write the movie header.
fileType: A UTI indicating the specific file format (e.g. AVFileTypeQuickTimeMovie for a QuickTime movie).
options: An integer whose bits specify options for the writing of the movie header. Can be AVMovieWritingTruncateDestinationToMovieHeaderOnly.
Error: If an error occurs writing the movie header, describes the nature of the failure.

Data references in the output movie header are adjusted to be relative to the destination URL. Note that modifications to instances of AVMutableMovie, to their constituent AVMutableMovieTracks, or to their collections of metadata are committed to storage when their movie headers are written.

6.4.23 Properties

6.4.24 canContainMovieFragments as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Indicates whether the movie file is capable of being extended by fragments.
Notes: The value of this property is true if an 'mvex' box is present in the 'moov' box. The 'mvex' box is necessary in order to signal the possible presence of later 'moof' boxes.
(Read only property)

6.4.25 containsMovieFragments as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Indicates whether the movie file is extended by at least one movie fragment.
Notes: The value of this property is YES if canContainMovieFragments is YES and at least one 'moof' box is present after the 'moov' box.
Available on OS X 10.11 or newer.
(Read only property)

6.4.26 Data as MemoryBlock

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The data block with which the instance of AVMovie was initialized; may be nil.
Notes: Available on OS X 10.11 or newer.
(Read only property)

6.4.27 defaultMediaDataStorage as AVMediaDataStorageMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The default storage container for media data added to a movie.
Notes: The value of this property is an AVMediaDataStorage object that indicates where sample data that is added to a movie should be written by default.
(Read only property)

6.4.28 URL as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The URL with which the instance of AVMovie was initialized; may be "".
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

MsgBox m.URL
```

Notes: (Read only property)

6.4.29 Constants

Constants

Constant	Value	Description
AVMovieWritingTruncateDestinationToMovieHeaderOnly	1	One of the options for writeMovieHeaderToURL. If set, writing the movie header will truncate all existi file and write a new movie header, thereby creating a A file type box will be written at the beginning of tl the movie header will remove any existing movie hea and write a new movie header, preserving any othe destination file was empty, a file type box will be wr the file. You would not want to use the AVMovieWritin MovieHeaderOnly option if you had written sample file using (for example) insertTimeRange with copy since that data would be lost.

6.5 class AVMovieTrackMBS

6.5.1 class AVMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: AVMovieTrack represents the tracks of audiovisual containers in a file that conforms to the QuickTime movie file format or to one of the related ISO base media file formats (such as MPEG-4).
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.m4v")
dim m as new AVMovieMBS(f, nil)

dim videotracks() as AVMovieTrackMBS = m.movieTracksWithMediaCharacteristic(AVFoundationMBS.AVMediaCharacteristicFrameBased)

for each videotrack as AVMovieTrackMBS in videotracks
dim s as CGSizeMBS = videotrack.naturalSize
MsgBox str(s.Width)+" x "+ str(s.Height)
next
```

Notes: Subclass of the AVAssetTrackMBS 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 / Real Studio Plugins, version 16.0pr5

6.5.2 Methods

6.5.3 available as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether this class is available.
Example:

```
if not AVMovieTrackMBS.available then
MsgBox "This application requires a newer OS X version."
end if
```

Notes: Returns true on OS X 10.10.

6.5.4 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.

6.5.5 Properties

6.5.6 alternateGroupID as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: An integer indicating the track as a member of a particular alternate group.
Notes: (Read only property)

6.5.7 mediaDataStorage as AVMediaDataStorageMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The storage container for media data added to a track.
Notes: The value of this property is an AVMediaDataStorage object that indicates the location at which media data inserted or appended to the track will be written.
(Read only property)

6.5.8 mediaDecodeTimeRange as CMTimeRangeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CMTimeRange indicating the range of decode times for the track's media.
Notes: (Read only property)

6.5.9 mediaPresentationTimeRange as CMTimeRangeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CMTimeRange indicating the range of presentation times for the track's media.
Notes: (Read only property)

6.6 class AVMutableMovieMBS

6.6.1 class AVMutableMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: AVMutableMovie adds to its immutable superclass, AVMovie, several categories of methods for editing QuickTime movie files, e.g. inserting and removing time ranges of media, adding and removing tracks, and modifying the metadata collections stored therein.
Notes: By default, after creating an AVMutableMovie the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

Available in OS X 10.11.
Subclass of the AVMovieMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.0pr5

6.6.2 Methods

6.6.3 addMutableTracksCopyingSettingsFromTracks(existingTracks() as AVMovieTrackMBS, options as Dictionary) as AVMutableMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Adds one or more empty tracks to the target movie, copying track settings from the source tracks.
Notes: existingTracks: An array of AVMovieTrack objects.
options: An dictionary object that contains keys for specifying options for the initialization of the new AVMutableMovieTrack objects. Currently no keys are defined; pass nil for default initialization behavior.
Returns An array of AVMutableMovieTrack objects; the index of a track in this array is the same as the index of its source track in the existingTracks array.
This method creates one or more empty tracks in the target movie and configures those tracks with settings (such as track userdata and metadata, width, height, and preferred volume) copied from the source tracks in the existingTracks array. Also, properties involving pairs of tracks (such as track references) are copied from the source tracks to the target tracks.

6.6.4 addMutableTrackWithMediaType(mediaType as String, track as AVMovieTrackMBS, options as Dictionary) as AVMutableMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Adds an empty track to the target movie.
Notes:

mediaType:	The media type of the new track (e.g. AVMediaTypeVideo for a video track).
track:	If you wish to transfer settings from an existing track, including track userdata and metadata, width, height, preferred volume, etc., pass a reference to an AVMovieTrack representing that track. Otherwise pass nil.

options: An Dictionary object that contains keys for specifying options for the initialization of the new AVMutableMovieTrack object. Currently no keys are defined; pass nil for default initialization behavior.
Returns An AVMutableMovieTrack object or nil in case of error.
The trackID of the newly added track is a property of the returned instance of AVMutableMovieTrack.

6.6.5 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The constructor to create blank new movie.
See also:

- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a Memoryblock/string.
Notes: data: A memoryblock containing a movie header.
options: An dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Raises exception in case of error.

You can use this method to operate on movie headers that are not stored in files. In general you should avoid loading an entire movie file with its media data into an instance of memoryblock!

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an memoryblock and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a Memoryblock/string.
Notes: data: A memoryblock containing a movie header.
options: An dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Raises exception in case of error.

You can use this method to operate on movie headers that are not stored in files. In general you should avoid loading an entire movie file with its media data into an instance of memoryblock!

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an memoryblock and then append sample buffers to any of its tracks, you must first set one of these properties to indicate

where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: File: An folderitem that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

Raises exception in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902

- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: File: An folderitem that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

Raises exception in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object without tracks (and therefore without media).
Notes: movie: If you wish to transfer settings from an existing movie (including movie userdata and metadata, preferred rate, preferred volume, etc.), pass a reference to an AVMovie object representing that movie. Otherwise pass nil. The userdata and metadata from the source movie may need to be converted if the format of that movie differs from fileType; you may wish to inspect the userdata or metadata of the receiver to ensure that important data was copied.
options: A dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined; pass nil for default initialization behavior.
Error: If an error occurs creating a movie, describes the nature of the failure.

Raises exception in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.11 Constructor(URL as String, Options as Dictionary = nil)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: URL: An URL that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.

Error: If an error occurs creating a movie, describes the nature of the failure. Optional.

Raises exception in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899
- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS) 904

6.6.12 Constructor(URL as String, Options as Dictionary = nil, byref error as NSErrorMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: URL: An URL that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure. Optional.

Raises exception in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.5 Constructor 899

- 6.6.6 Constructor(Data as MemoryBlock, Options as Dictionary = nil) 899
- 6.6.7 Constructor(Data as MemoryBlock, Options as Dictionary = nil, byref error as NSErrorMBS) 900
- 6.6.8 Constructor(File as FolderItem, Options as Dictionary = nil) 901
- 6.6.9 Constructor(File as FolderItem, Options as Dictionary = nil, byref error as NSErrorMBS) 902
- 6.6.10 Constructor(Movie as AVMovieMBS = nil, Options as Dictionary = nil, byref error as NSErrorMBS) 903
- 6.6.11 Constructor(URL as String, Options as Dictionary = nil) 903

6.6.13 insertEmptyTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Adds an empty time range to the target movie.
Notes: timeRange: The time range to be made empty. Note that you cannot add empty time ranges to the end of a movie.

6.6.14 insertTimeRange(timeRange as CMTimeRangeMBS, asset as AVAssetMBS, atTime as CMTimeMBS, copySampleData as Boolean, byref Error as NSErrorMBS) as boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Inserts all the tracks of a timeRange of an asset into a movie.
Notes:

timeRange:	The time range of the asset to be inserted.
asset:	An AVAsset object indicating the source of the inserted media. Only instances of AVURLAsset and AVComposition are supported. Must not be nil.

startTime: The time in the target movie at which the media is to be inserted.
copySampleData: A boolean value that indicates whether sample data is to be copied from the source to the destination during edits. If true, the sample data is written to the location specified by the track property mediaDataStorage if non-nil, or else by the movie property defaultMediaDataStorage if non-nil; if both are nil, the method will fail and return false. If false, sample data will not be written and sample references to the samples in their original container will be added as necessary.
Note that in this case, this method will fail if the source AVAsset is not able to provide sample reference information for the original container.
Error: If the insertion fails, an error object that describes the nature of the failure.

Returns A boolean value that indicates the success of the insertion.

This method may add new tracks to the target movie to ensure that all tracks of the asset are represented in the inserted timeRange.
Existing content at the specified startTime will be pushed out by the duration of timeRange.

6.6.15 metadata as AVMetadataItemMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A collection of metadata stored by the movie.
Notes: The value of this property is an array of AVMetadataItem objects representing the collection of metadata stored by the movie.

6.6.16 movieWithData(Data as MemoryBlock, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a Memoryblock/string.
Notes: data: A memoryblock/string containing a movie header.
options: An dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

You can use this method to operate on movie headers that are not stored in files. In general you should avoid loading an entire movie file with its media data into an instance of memoryblock!

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an memoryblock and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.17 movieWithData(Data as String, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS 907

6.6.17 movieWithData(Data as String, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a Memoryblock/string.
Notes: data: A memoryblock/string containing a movie header.
options: An dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

You can use this method to operate on movie headers that are not stored in files. In general you should avoid loading an entire movie file with its media data into an instance of memoryblock!

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an memoryblock and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.
See also:

- 6.6.16 movieWithData(Data as MemoryBlock, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS 906

6.6.18 movieWithFile(File as FolderItem, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: File: An folderitem that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

6.6.19 movieWithSettingsFromMovie(Movie as AVMovieMBS, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object without tracks (and therefore without media).
Notes: movie: If you wish to transfer settings from an existing movie (including movie userdata and metadata, preferred rate, preferred volume, etc.), pass a reference to an AVMovie object representing that movie. Otherwise pass nil. The userdata and metadata from the source movie may need to be converted if the format of that movie differs from fileType; you may wish to inspect the userdata or metadata of the receiver to ensure that important data was copied.
options: A dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined; pass nil for default initialization behavior.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMovieTrack's mediaDataStorage property will be nil. If you want to create an AVMutableMovie from an NSData object and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

6.6.20 movieWithURL(URL as String, Options as Dictionary = nil, byref Error as NSErrorMBS) as AVMovieMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Creates an AVMutableMovie object from a movie header stored in a QuickTime movie file or ISO base media file.
Notes: URL: An URL that specifies a file containing a movie header.
options: An Dictionary object that contains keys for specifying options for the initialization of the AVMutableMovie object. Currently no keys are defined.
Error: If an error occurs creating a movie, describes the nature of the failure.

Returns An AVMutableMovie object or nil in case of error.

By default, the defaultMediaDataStorage property will be nil and each associated AVMutableMovieTrack's mediaDataStorage property will be nil.
If you want to create an AVMutableMovie from a file and then append sample buffers to any of its tracks, you must first set one of these properties to indicate where the sample data should be written.

6.6.21 mutableMovieTracks as AVMutableMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The tracks in a mutable movie.
Notes: The value of this property is an array of tracks the mutable movie contains; the tracks are of type AVMutableMovieTrack.

6.6.22 mutableMovieTracksWithMediaCharacteristic(mediaCharacteristic as string) as AVMutableMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVMutableMovieTracks of the asset that present media with the specified characteristic.
Notes: mediaCharacteristic: The media characteristic according to which the receiver filters its AVMutableMovieTracks. (Media characteristics are defined in AVMediaFormat.h)
An array of AVMutableMovieTracks; may be empty if no tracks with the specified characteristic are available.
Becomes callable without blocking when the key "tracks" has been loaded.

6.6.23 mutableMovieTracksWithMediaType(mediaType as string) as AVMutableMovieTrackMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an array of AVMutableMovieTracks of the asset that present media of the specified media type.
Notes: mediaType: The media type according to which the receiver filters its AVMutableMovieTracks. (Media types are defined in AVMediaFormat.h)
An array of AVMutableMovieTracks; may be empty if no tracks of the specified media type are available.
Becomes callable without blocking when the key "tracks" has been loaded.

6.6.24 mutableMovieTrackWithTrackID(ID as Integer) as AVMutableMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides an instance of AVMutableMovieTrack that represents the track of the specified trackID.
Notes: ID: The trackID of the requested AVMutableMovieTrack.

An instance of AVMutableMovieTrack; may be nil if no track of the specified trackID is available.
Becomes callable without blocking when the key "tracks" has been loaded.

6.6.25 mutableTrackCompatibleWithTrack(track as AVAssetTrackMBS) as AVMutableMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Provides a reference to a track of a mutable movie into which any time range of an AVAssetTrack can be inserted (via insertTimeRange:ofTrack).
Notes: track: A reference to the AVAssetTrack from which a time range may be inserted.
Returns An AVMutableMovieTrack that can accommodate the insertion.
If no such track is available, the result is nil. A new track of the same media type as the AVAssetTrack can be created via addMutableTrackWithMediaType, and this new track will be compatible.

For best performance, the number of tracks in a movie should be kept to a minimum, corresponding to the number for which media data must be presented in parallel. If media data of the same type is to be presented serially, even from multiple assets, a single track of that media type should be used. This method, mutableTrackCompatibleWithTrack, can help the client to identify an existing target track for an insertion.

6.6.26 removeTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Removes a specified time range from a movie.
Notes: timeRange: The time range to be removed.

6.6.27 removeTrack(track as AVMovieTrackMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Removes a track from the target movie.
Notes: track: The track to be removed.

6.6.28 scaleTimeRange(timeRange as CMTimeRangeMBS, duration as CMTimeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Changes the duration of a time range of a movie.
Notes: timeRange: The time range to be scaled.

duration: The new duration of the time range.

6.6.29 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Sets the metadata.

6.6.30 Properties

6.6.31 defaultMediaDataStorage as AVMediaDataStorageMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The default storage container for media data added to a movie.
Notes: The value of this property is an AVMediaDataStorage object that indicates where sample data that is added to a movie should be written, for any track for whose mediaDataStorage property is nil.
(Read and Write property)

6.6.32 interleavingPeriod as CMTimeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CMTime that indicates the duration for interleaving runs of samples of each track.
Notes: The default interleaving period is 0.5 seconds.
(Read and Write property)

6.6.33 modified as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether a movie has been modified.
Notes: The value of this property is a boolean that indicates whether the AVMutableMovie object has been modified since it was created, was last written, or had its modified state cleared via assignment Modified = false.
(Read and Write property)

6.6.34 preferredRate as Double

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The natural rate at which the movie is to be played; often but not always 1.0.
Notes: (Read and Write property)

6.6.35 preferredTransform as CMTimeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGAffineTransform indicating the transform specified in the movie's storage container as the preferred transformation of the visual media data for display purposes; the value is often but not always CGAffineTransformIdentity.
Notes: (Read and Write property)

6.6.36 preferredVolume as Double

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The preferred volume of the audible media data of the movie; often but not always 1.0.
Notes: (Read and Write property)

6.6.37 timescale as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that contain a 'moov' atom, such as QuickTime Movie files, specifies the time scale of the movie.
Notes: The default movie time scale is 600. In certain cases, you may want to set this to a different value. For instance, a movie that contains a single audio track should typically have the movie time scale set to the media time scale of that track.

This property should be set on a new empty movie before any edits are performed on the movie.
(Read and Write property)

6.7 class AVMutableMovieTrackMBS

6.7.1 class AVMutableMovieTrackMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: AVMutableMovieTrack provides the track-level editing interface of an AVMutableMovie.
Notes: Media can be inserted into a movie track and other editing operations performed via an instance of this class.
Available on OS X 10.11.
Subclass of the AVMovieTrackMBS 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 / Real Studio Plugins, version 16.0pr5

6.7.2 Methods

6.7.3 addTrackAssociationToTrack(movieTrack as AVMovieTrackMBS, trackAssociationType as String)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Establishes a track association of a specific type between two tracks.
Notes: movieTrack: An AVMovieTrack object that is to be associated with the receiver.
trackAssociationType: The type of track association to add between the receiver and the specified movieTrack (for instance, AVTrackAssociationTypeChapterList).

6.7.4 Constructor

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The private constructor.

6.7.5 insertEmptyTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Adds an empty time range to the target track.
Notes: timeRange: The time range to be made empty. Note that you cannot add empty time ranges to the end of a track.

6.7.6 insertTimeRange(timeRange as CMTimeRangeMBS, assetTrack as AVAssetTrackMBS, atTime as CMTimeMBS, copySampleData as Boolean, byref Error as NSErrorMBS) as boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Inserts a portion of an AVAssetTrack object into the target movie.
Notes: timeRange: The time range from the track from which media is to be inserted.
track: An AVAssetTrack object indicating the source of the inserted media. Only AVAssetTracks of AVURLAssets and AVCompositions are supported. Must not be nil.
startTime: The time in the target track at which the media is to be inserted.
copySampleData: A boolean value that indicates whether sample data is to be copied form the source to the destination during edits.

If true, the sample data is written to the file specified by the track property mediaDataStorage if non-nil, or else by the movie property defaultMediaDataStorage if non-nil; if both are nil, the method will fail and return NO.
If false, sample data will not be written and sample references to the samples in their original container will be added
as necessary. Note that in this case, this method will fail if the original samples are fragmented.

Error: If the insertion fails, describes the nature of the failure.

Returns a boolean value that indicates the success of the insertion.

6.7.7 metadata as AVMetadataItemMBS()

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A collection of metadata stored by the track.
Notes: The value of this property is an array of AVMetadataItem objects representing the collection of metadata stored by the track.

6.7.8 removeTimeRange(timeRange as CMTimeRangeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Removes a specified time range from a track.
Notes: timeRange: The time range to be removed.

6.7.9 removeTrackAssociationToTrack(movieTrack as AVMovieTrackMBS, trackAssociationType as String)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Removes a track association of a specific type between two tracks.
Notes: movieTrack: An AVMovieTrack object that is associated with the receiver.
trackAssociationType: The type of track association to remove between the receiver and the specified movieTrack (for instance, AVTrackAssociationTypeChapterList).

6.7.10 scaleTimeRange(timeRange as CMTimeRangeMBS, duration as CMTimeMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Changes the duration of a time range of a track.
Notes: timeRange: The time range to be scaled.
duration: The new duration of the time range.

6.7.11 setMetadata(items() as AVMetadataItemMBS)

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Sets the metadata items.

6.7.12 Properties

6.7.13 alternateGroupID as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: An integer indicating the track as a member of a particular alternate group.
Notes: (Read and Write property)

6.7.14 cleanApertureDimensions as CGSizeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGSize indicating the clean aperture dimensions of the track.
Notes: (Read and Write property)

6.7.15 Enabled as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A boolean value indicating whether the track is enabled by default for presentation.
Notes: (Read and Write property)

6.7.16 encodedPixelsDimensions as CGSizeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGSize indicating the dimensions encoded pixels dimensions of the track.
Notes: (Read and Write property)

6.7.17 extendedLanguageTag as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The language tag associated with the track.
Notes: The value of this property is an IETF BCP 47 (RFC 4646) language identifier indicating the language tag associated with the track; may be "" if no language tag is indicated.
(Read and Write property)

6.7.18 hasProtectedContent as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether a track contains protected content.
Notes: The value of this property is a boolean that indicates whether the track contains protected content.
(Read only property)

6.7.19 languageCode as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The language associated with the track.
Notes: The value of this property is an ISO 639-2/T language code indicating the language associated with the track; may be "" if no language is indicated.
(Read and Write property)

6.7.20 layer as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The layer level of the visual media data of the track.
Notes: (Read and Write property)

6.7.21 mediaDataStorage as AVMediaDataStorageMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The storage container for media data added to a track.
Notes: The value of this property is an AVMediaDataStorage object that indicates the location at which media data inserted or appended to the track will be written.
(Read and Write property)

6.7.22 Modified as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: Whether a track has been modified.
Notes: The value of this property is a boolean that indicates whether the AVMutableMovieTrack object has been modified since it was created, was last written, or had its modified state cleared via an assignment: Modified = false.
(Read and Write property)

6.7.23 naturalSize as CGSizeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGSize indicating the dimensions at which the visual media data of the track should be displayed.
Notes: (Read and Write property)

6.7.24 preferredMediaChunkAlignment as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that support media chunk alignment, the boundary for media chunk alignment (in bytes).
Notes: The default value is 0, which means that the receiver will choose an appropriate default value.
A value of 1 implies that no padding should be used to achieve a particular chunk alignment. It is an error

to set a negative value for chunk alignment.
(Read and Write property)

6.7.25 preferredMediaChunkDuration as CMTimeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that support media chunk durations, the maximum duration to be used for each chunk of sample data written to the file.
Notes: The total duration of the samples in a chunk will be no greater than this preferred chunk duration, or the duration of a single sample if the sample's duration is greater than this preferred chunk duration.

The default value is kCMTimeInvalid, which means that the receiver will choose an appropriate default chunk duration. It is an error to set a chunk duration that is negative or non-numeric.

This property and preferredMediaChunkSize, which also specifies the size of a chunk, are mutually exclusive. Thus, if both properties are set, the last setting is honored.
(Read and Write property)

6.7.26 preferredMediaChunkSize as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that support media chunk sizes, the maximum size (in bytes) to be used for each chunk of sample data written to the file.
Notes: The total size of the samples in a chunk will be no larger than this preferred chunk size, or the size of a single sample if the sample is larger than this preferred chunk size.

The default value is 0, which means that the receiver will choose an appropriate default chunk size. It is an error to set a negative chunk size.

This property and preferredMediaChunkDuration, which also specifies the size of a chunk, are mutually exclusive.
Thus, if both properties are set, the last setting is honored.
(Read and Write property)

6.7.27 preferredTransform as CGAffineTransformMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGAffineTransform indicating the transform specified in the track's storage container as the preferred transformation of the visual media data for display purposes; the value is often but not always CGAffineTransformIdentity.
Notes: (Read and Write property)

6.7.28 preferredVolume as Double

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: The preferred volume of the audible media data of the track; often but not always 1.0.
Notes: (Read and Write property)

6.7.29 productionApertureDimensions as CGSizeMBS

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: A CGSize indicating the production aperture dimensions of the track.
Notes: (Read and Write property)

6.7.30 sampleReferenceBaseURL as String

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that support writing sample references, such as QuickTime Movie files, specifies the base URL that sample references are relative to; may be nil.
Notes: If the value of this property can be resolved as an absolute URL, the sample locations written to the file when appending sample references to this track will be relative to this URL. The URL must point to a location contained by any common parent directory of the locations that will be referenced. For example, setting the sampleReferenceBaseURL property to "file:///Users/johnappleseed/Movies/" and appending sample buffers that refer to "file:///Users/johnappleseed/Movies/data/movie1.mov" will cause the sample reference "data/movie1.mov" to be written to the movie file.

If the value of the property cannot be resolved as an absolute URL or if it points to a location that is not contained by any common parent directory of the locations that will be referenced, the location will be written unmodified.

The default value is "", which means that the location will be written unmodified.
(Read and Write property)

6.7.31 timescale as Integer

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: For file types that contain a 'moov' atom, such as QuickTime Movie files, specifies the time scale of the track's media.
Notes: The default media time scale is 0.
This property should be set on a new empty track before any edits are performed on the track.
(Read and Write property)

Chapter 7

Image Capture

7.1 control DesktopIKCameraDeviceViewControlMBS

7.1.1 control DesktopIKCameraDeviceViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Camera Device View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.1.2 Properties

7.1.3 View as IKCameraDeviceViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The camera device view used in this control.
Notes: (Read only property)

7.1.4 Events

7.1.5 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.1.6 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.

7.1.7 DidDownloadFile(CameraFile as ICCameraFileMBS, URL as string, File as folderItem, data as MemoryBlock, error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent for each file that gets downloaded.
Notes: Based on the IKCameraDeviceViewDisplayMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.1.8 DidEncounterError(Error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent every time the camera device reports an error.

7.1.9 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.

7.1.10 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.

7.1.11 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.

7.1.12 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.

7.1.13 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.

7.1.14 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.

7.1.15 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.

7.1.16 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.

7.1.17 SelectionDidChange

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.

7.1.18 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.

7.2 control DesktopIKDeviceBrowserViewControlMBS

7.2.1 control DesktopIKDeviceBrowserViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Device Browser View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.2.2 Properties

7.2.3 View as IKDeviceBrowserViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The device browser view used in this control.
Notes: (Read only property)

7.2.4 Events

7.2.5 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.2.6 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.

7.2.7 DidEncounterError(error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent every time the device browser reports an error.

7.2.8 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.

7.2.9 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.

7.2.10 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.

7.2.11 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.

7.2.12 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.

7.2.13 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.

7.2.14 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.

7.2.15 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.

7.2.16 SelectionDidChange(device as ICDeviceMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.
Notes: The device may be a ICCameraDeviceMBS or a ICScannerDeviceMBS.

7.2.17 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.

7.3 control DesktopIKScannerDeviceViewControlMBS

7.3.1 control DesktopIKScannerDeviceViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Scanner Device View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.3.2 Properties

7.3.3 View as IKScannerDeviceViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The scanner view used in this control.
Notes: For Xojo with Cocoa target.
(Read only property)

7.3.4 Events

7.3.5 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.3.6 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.

7.3.7 DidEncounterError(error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: This event is sent every time the scanner device reports an error.

7.3.8 DidScanToBandData(data as ICScannerBandDataMBS, scanInfo as Dictionary, error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: For memory a based transfer this event is sent for every time an image band of data was scanned.
Notes: The 'data' parameter describes the scanned image data. Note that rotation/cropping/image adjustments are not applied yet. The 'scanInfo' parameter contains additional information (rotation angle, ...) that should be applied once the scan is completed.

7.3.9 DidScanToURL(url as String, file as FolderItem, fileData as MemoryBlock, error as NSErrorMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: For file based transfer this event is sent for each image that gets scanned.
Notes: Based on the IKScannerDeviceViewTransferMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.3.10 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.

7.3.11 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.

7.3.12 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.

7.3.13 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.

7.3.14 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.

7.3.15 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.

7.3.16 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.

7.3.17 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.

7.3.18 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.

7.4 class ICCameraDeviceMBS

7.4.1 class ICCameraDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICCameraDeviceMBS is a concrete subclass of ICDeviceMBS class.
Notes: ICDeviceBrowserMBS creates instances of this class.
Subclass of the ICDeviceMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.4.2 Methods

7.4.3 cancelDelete

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Cancels the current delete operation started by sending a requestDeleteFiles.

7.4.4 cancelDownload

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Cancels the current download operation.

7.4.5 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.4.6 contents as ICCameraItemMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Contents of the camera.
Notes: The structure of the elements in this array will reflect the folder structure of the storage reported by the camera. Each item in this array will correspond to a storage on the camera.

7.4.7 filesOfType(fileUTType as string) as ICCameraFileMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method returns an array of files on the camera of type fileType.
Notes: The fileType string is one of the following Uniform Type Identifier strings: kUTTypeImage, kUTTypeMovie, kUTTypeAudio, or kUTTypeData.
See UTTypeMBS module.

7.4.8 ICCameraDeviceCanAcceptPTPCommands as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can accept PTP commands.

7.4.9 ICCameraDeviceCanDeleteAllFiles as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can delete all files in a single operation while it is connected.

7.4.10 ICCameraDeviceCanDeleteOneFile as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can delete a file at a time while it is connected.

7.4.11 ICCameraDeviceCanReceiveFile as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the host can upload files to the camera.

7.4.12 ICCameraDeviceCanSyncClock as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can synchronize its date and time with that of the host computer.

7.4.13 ICCameraDeviceCanTakePicture as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can capture a picture while it is connected, if the client sends a requestTakePicture message to it.

7.4.14 ICCameraDeviceCanTakePictureUsingShutterReleaseOnCamera as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a camera.
Notes: Indicates that the camera can capture a picture while it is connected, if the user presses the shutter release on the camera.

7.4.15 ICDeleteAfterSuccessfulDownload as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key should be a boolean value. If this value is true, the file will be deleted from the device after it is succcessfully downloaded.

7.4.16 ICDownloadsDirectoryURL as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key should be an CFURLMBS referencing a writable directory.
The downloaded files will be saved in that directory.

7.4.17 ICDownloadSidecarFiles as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key should be a boolean value. If this value is true, all sidecar files will be downloaded along with the media file.

7.4.18 ICOverwrite as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key should be a boolean value. If this value is true, the downloaded file will overwrite an existing file with the same name and extension.

7.4.19 ICSaveAsFilename as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key should be a string containing the name to be used for the downloaded file.

7.4.20 ICSavedAncillaryFiles as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key will be an array containing names of files associated with the primary file that is downloaded. The options dictionary returned in didDownloadFile may have this key.

7.4.21 ICSavedFilename as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the keys for options dictionary.
Notes: The value for this key will be a string containing the actual name of the saved file. The options dictionary returned in didDownloadFile will have this key.

7.4.22 mediaFiles as ICCameraFileMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The property mediaFiles represents all image, movie and audio files on the camera.
Notes: These files are returned as a single array without regard to the folder hierarchy used to store these files on the camera.

If no files show up, try again a second later as the framework may not yet have the list loaded.

7.4.23 requestDeleteFiles(files() as ICCameraFileMBS)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Deletes files.

7.4.24 requestDisableTethering

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Send this message to disable tethered capture on the camera device if the camera has the 'ICCameraDeviceCanTakePicture' capability and if your process has already sent a 'requestEnableTethering' to it.

7.4.25 requestDownloadFile(file as ICCameraFileMBS, options as dictionary = nil)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Download a file from the camera. Please refer to the top of this header for information about the options.
Notes: Calls cameraDeviceDidDownloadFile event later.
The content of error returned should be examined to determine if the request completed successfully.

7.4.26 requestEnableTethering

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Send this message to enable tethered capture on the camera device if the camera has the 'ICCameraDeviceCanTakePicture' capability.

7.4.27 requestReadDataFromFile(file as ICCameraFileMBS, offset as UInt64, Length as UInt64)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method asynchronously reads data of a specified length from a specified offset.
Notes: Calls later ImageCaptureEventsMBS.cameraDeviceDidReadData event.
The content of error returned should be examined to determine if the request completed successfully.

7.4.28 requestSendPTPCommand(command as MemoryBlock, dataOut as MemoryBlock)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method asynchronously sends a PTP command to a camera.
Notes: The content of error returned should be examined to determine if the request completed successfully.

7.4.29 requestSyncClock

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Synchronize camera's clock with the computer's clock.
Notes: You should send this request only if the camera has the 'ICCameraDeviceCanSyncClock' capability.

7.4.30 requestTakePicture

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Capture a new image using the camera, the camera capabilities include 'ICCameraDeviceCanTakePicture'.
Notes: You MUST send 'requestEnableTethering' message to the camera before sending 'requestTakePicture' message.

7.4.31 requestUploadFile(file as folderitem, options as dictionary = nil)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Upload a file at fileURL to the camera.
Notes: The options dictionary is not used in this version.
Calls later ImageCaptureEventsMBS.cameraDeviceDidUploadFile event.

The content of error returned should be examined to determine if the request completed successfully.

7.4.32 Properties

7.4.33 batteryLevel as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates the battery charge level.
Notes: Its value ranges from 0 to 100.
(Read only property)

7.4.34 batteryLevelAvailable as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the device has reported battery charge levelÔøº.
Notes: (Read only property)

7.4.35 contentCatalogPercentCompleted as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates the percentage of content cataloging completed on the device.
Notes: Its value ranges from 0 to 100.
(Read only property)

7.4.36 isAccessRestrictedAppleDevice as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Set to true if the device is made by Apple and is pass-coded locked and connected to an untrusted host.
Notes: (Read only property)

7.4.37 mountPoint as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Filesystem mount point for a device with transportType of ICTransportTypeMassStorage.
Notes: This will be ”” for all other devices.
(Read only property)

7.4.38 tetheredCaptureEnabled as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This property is set to YES when tethered capture is enabled on the device.
Notes: Use ’requestEnableTethering’ and ’requestDisableTethering’ to enable or disable tethered capture on the device.
(Read only property)

7.4.39 timeOffset as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates the time offset, in seconds, between the camera’s clock and the computer’s clock.
Notes: This value is positive if the camera’s clock is ahead of the computer’s clock. This property should be ignored if the camera’s capabilities property does not contain ICCameraDeviceCanSyncClock.
(Read only property)

7.5 class ICCameraFileMBS

7.5.1 class ICCameraFileMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This class represents a file on an ICCameraDevice object.
Notes: Subclass of the ICCameraItemMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.5.2 Methods

7.5.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.5.4 sidecarFiles as ICCameraFileMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Returns array of sidecar files.
Notes: This property is an empty array if there are no sidecar files associated with this file. Otherwise it is an array of ICCameraFile instances of sidecar files associated with this file. An example of a sidecar file is a file with the same base name as this file and having an extension XMP.

7.5.5 Properties

7.5.6 Duration as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Duration of audio/video file in seconds.
Notes: (Read only property)

7.5.7 FileSize as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Size of file in bytes.
Notes: (Read only property)

7.5.8 Orientation as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Desired orientation of image to use when it is downloaded.
Notes: This property is set to ICEXIFOrientation1 initially. If the format of this file supports EXIF orientation tag, then this property will be updated to match the value of that tag, when the thumbnail or metadata for this file is received.

Possible values:

ICEXIFOrientation1	1	Normal
ICEXIFOrientation2	2	Flipped horizontally
ICEXIFOrientation3	3	Rotated 180¬∞
ICEXIFOrientation4	4	Flipped vertically
ICEXIFOrientation5	5	Rotated 90¬∞ CCW and flipped vertically
ICEXIFOrientation6	6	Rotated 90¬∞ CCW
ICEXIFOrientation7	7	Rotated 90¬∞ CW and flipped vertically
ICEXIFOrientation8	8	Rotated 90¬∞ CW

(Read and Write property)

7.6 class ICCameraFolderMBS

7.6.1 class ICCameraFolderMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This class represents a folder on an ICCameraDevice object.
Notes: Subclass of the ICCameraItemMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.6.2 Methods

7.6.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.6.4 contents as ICCameraItemMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A list of items contained by this folder.

7.7 class ICCameraItemMBS

7.7.1 class ICCameraItemMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICCameraItem is an abstract class that represents an item in an ICCameraDevice object.
Notes: ICCameraDevice object creates instances of two concrete subclasses of ICCameraItem: ICCameraFolder and ICCameraFile.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.7.2 Methods

7.7.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.7.4 Properties

7.7.5 addedAfterContentCatalogCompleted as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This property is set if the file is captured on the device after the device's content is fully enumerated.
Notes: This does not apply to files added as a result of adding a new store to the device.
(Read only property)

7.7.6 CreationDate as Date

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creation date of this file.
Notes: This information is usually the same as the EXIF creation date.
(Read only property)

7.7.7 CreationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Creation date of this file.
Notes: This information is usually the same as the EXIF creation date.
(Read only property)

7.7.8 Device as ICCameraDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Parent device of this folder.
Notes: (Read only property)

7.7.9 FileSystemPath as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The file system path of the item for items on a device with transportType of ICTransportTypeMassStorage.
Notes: (Read only property)

7.7.10 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.7.11 InTemporaryStore as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if this folder is in a temporary store.
Notes: A temporary store may be used by the device when images are captures on the device when it is tethered to the computer.
(Read only property)

7.7.12 largeThumbnailIfAvailable as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Large thumbnail for the item if one is readily available.
Notes: Value is a CGImageMBS.
If one is not readily available, accessing this property will send a message to the device requesting a thumbnail for the file. The ImageCaptureEventsMBS subclass will be notified via event cameraDeviceDidReceiveThumbnailForItem.
(Read only property)

7.7.13 Locked as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates the protection state of this item.
Notes: It is locked if the storage card in the camera is locked.
(Read only property)

7.7.14 MetadataIfAvailable as Dictionary

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Metadata for the file if one is readily available.
Notes: If one is not readily available, accessing this property will send a message to the device requesting a thumbnail for the file. The ImageCaptureEventsMBS subclass will be notified via event cameraDeviceDidReceiveMetadataForItem.
(Read only property)

7.7.15 ModificationDate as Date

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Modification date of this file.
Notes: This information is usually the same as the EXIF modification date.
(Read only property)

7.7.16 ModificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: Modification date of this file.
Notes: This information is usually the same as the EXIF modification date.
(Read only property)

7.7.17 Name as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Name of this file or folder.
Notes: (Read only property)

7.7.18 ParentFolder as ICCameraFolderMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Parent folder of this folder. The root folder's parentFolder is nil.
Notes: (Read only property)

7.7.19 ptpObjectHandle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: PTP object handle value if the item is on a camera that uses PTP protocol.
Notes: The value of this property is set to 0 if the camera does not use PTP protocol.
(Read only property)

7.7.20 Raw as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the file is a raw image file.
Notes: (Read only property)

7.7.21 thumbnailIfAvailable as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Thumbnail for the item if one is readily available.
Notes: Value is a CGImageMBS.
If one is not readily available, accessing this property will send a message to the device requesting a

thumbnail for the file. The ImageCaptureEventsMBS subclass will be notified via event cameraDeviceDidReceiveThumbnailForItem.
(Read only property)

7.7.22 UserData as Dictionary

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A mutable dictionary to store arbitrary key-value pairs associated with a camera item object.
Notes: This can be used by view objects that bind to this object to store "house-keeping" information.
In Xojo, please query dictionary, modify it and assign back to this property.
(Read and Write property)

7.7.23 UTI as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Item UTI.
Notes: This is an Uniform Type Identifier string. It is one of: kUTTypeFolder, kUTTypeImage, kUTTypeMovie, kUTTypeAudio, or kUTTypeData.
See UTTypeMBS module.
(Read only property)

7.8 class ICDeviceBrowserMBS

7.8.1 class ICDeviceBrowserMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The ICDeviceBrowser object is used to find devices such as digital cameras and scanners that are supported by Image Capture.
Notes: These device may be directly attached to the USB or FireWire bus on the host computer, shared by other computers, or available over a TCP/IP network. This object communicates with an Image Capture agent process asynchronously to accomplish this.
Blog Entries

- MBS Xojo Plugins, version 19.0pr8
- MBS Xojo / Real Studio Plugins, version 16.4pr1
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.8.2 Methods

7.8.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The constructor.

7.8.4 Destructor

Plugin Version: 16.4, Platform: macOS, Targets: Desktop only.

Function: The destructor.

7.8.5 devices as ICDeviceMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: All devices found by the browser.
Notes: This property will change as devices appear and disappear. This array is empty before the first invocation of the deviceBrowserDidAddDevice event.

7.8.6 Start

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This message tells the receiver to start looking for devices.
Notes: Please use ImageCaptureEventsMBS class to receive events.

7.8.7 Stop

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method tells the receiver to stop looking for devices.
Notes: This will free all device instances that are not in use.

7.8.8 Properties

7.8.9 browsedDeviceTypeMask as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The device type mask.
Notes: A mask whose set bits indicate the type of device(s) being browsed after the receiver receives the start message. This property can be changed while the browser is browsing for devices. This property can be constructed by OR'd values of ICDeviceTypeMask with values of ICDeviceLocationTypeMask.
(Read and Write property)

7.8.10 Browsing as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the device browser is browsing for devices.
Notes: (Read only property)

7.8.11 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.8.12 preferredDevice as ICDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method returns a device object that should be selected by the client application when it is launched.
Notes: If the client application that calls this method is the auto-launch application associated with a device and that device is the last device attached (through USB, FireWire or network), then that device will be the preferred device. The best place to call this method is in the event deviceBrowserDidAddDevice, if the "moreComing" parameter passed to the delegate is false; or in the event deviceBrowserDidEnumerateLocalDevices.
(Read only property)

7.8.13 Events

7.8.14 DeviceDidChangeName(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent if the name of a device changes.
Notes: This happens if the device module overrides the default name of the device reported by the device's transport layer, or if the name of the filesystem volume mounted by the device is changed by the user.

7.8.15 DeviceDidChangeSharingState(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the sharing state of a device has changes.
Notes: Any Image Capture client application can choose to share the device over the network using the sharing or webSharing facility in Image Capture.

7.8.16 DidAddDevice(device as ICDeviceMBS, moreComing as boolean)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to inform that a device has been added.
Notes: If several devices are found during the initial search, then this event is sent once for each device with the value of 'moreComing' set to true in each event except the last one.

7.8.17 DidEnumerateLocalDevices

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent after the device browser completes sending deviceBrowser:didAddDevice event for all local devices.
Notes: Detecting locally connected devices (USB and FireWire devices) is faster than detecting devices connected using a network protocol. An Image Capture client application may use this event to update its user interface to let the user know that it has completed looking for locally connected devices and then start looking for network devices.

7.8.18 DidRemoveDevice(device as ICDeviceMBS, moreGoing as boolean)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to inform that a device has been removed.
Notes: If several devices are removed at the same time, then this event is sent once for each device with the value of 'moreGoing' set to true in each event except the last one.

7.8.19 RequestsSelectDevice(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an event that occurred on the device may be of interest to the client application.
Notes: In Mac OS X 10.6, this event is sent when a button is pressed on a device and the current application is the target for that button press. In the case of the button-press event, if a session is open on the device, this event will not be sent, instead the deviceDidReceiveButtonPress event is sent.

7.9 class ICDeviceMBS

7.9.1 class ICDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICDevice is an abstract class that represents a device supported by Image Capture.
Notes: ImageCaptureCore defines two concrete subclasses of ICDeviceMBS, ICCameraDeviceMBS and ICScannerDeviceMBS. ICDeviceBrowserMBS creates instances of these two subclasses to represent cameras and scanners it finds.
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 20.5pr10

7.9.2 Methods

7.9.3 capabilities as Variant()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The capabilities of the device as reported by the device module.

7.9.4 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.9.5 ICButtonTypeCopy as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the "Copy" button on the device was pressed.

7.9.6 ICButtonTypeMail as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the ”Mail” button on the device was pressed.

7.9.7 ICButtonTypePrint as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the ”Print” button on the device was pressed.

7.9.8 ICButtonTypeScan as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the ”Scan” button on the device was pressed.

7.9.9 ICButtonTypeTransfer as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the ”Transfer” button on the device was pressed.

7.9.10 ICButtonTypeWeb as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to identify button-press on a device.
Notes: Indicates that the ”Web” button on the device was pressed.

7.9.11 ICDeviceCanEjectOrDisconnect as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used to describe capabilities of a device.
Notes: Indicates either the device is mounted as a mass-storage volume and can be ejected or the it is a remote device with an active connection that can be disconnected.

7.9.12 ICDeviceLocationDescriptionBluetooth as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This description is returned for locationDescription property of a device connected via Bluetooth.

7.9.13 ICDeviceLocationDescriptionFireWire as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This description is returned for locationDescription property of a device connected to a FireWire port.

7.9.14 ICDeviceLocationDescriptionMassStorage as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This description is returned for locationDescription property of a device that is mounted as a mass-storage volume.

7.9.15 ICDeviceLocationDescriptionUSB as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This description is returned for locationDescription property of a device connected to a USB port.

7.9.16 ICLocalizedStatusNotificationKey as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used for device status notifications.
Notes: Key for a localized notification string.

7.9.17 ICStatusCodeKey as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used for device status notifications.
Notes: One of values defined in ICReturnCode.

7.9.18 ICStatusNotificationKey as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the constants used for device status notifications.
Notes: Key for a non-localized notification string.

7.9.19 ICTransportTypeBluetooth as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates that the device uses Bluetooth transport.

7.9.20 ICTransportTypeFireWire as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates that the device uses FireWire transport.

7.9.21 ICTransportTypeMassStorage as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates that the device use mounts as a mass-storage volume.

7.9.22 ICTransportTypeTCPIP as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates that the device uses TCP/IP transport.
Notes: These devices are discovered using Bonjour.

7.9.23 ICTransportTypeUSB as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates that the device uses USB transport.

7.9.24 requestCloseSession

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This message requests to close a previously opened session on this device.
Notes: This request is completed when the ImageCaptureEventsMBS subclass receives a deviceDidCloseSessionWithError event.

7.9.25 requestEjectOrDisconnect

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Eject the media if permitted by the device, or disconnect from a remote device.

7.9.26 requestOpenSession

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This message requests to open a session on the device.
Notes: A client MUST open a session on a device in order to use the device.
This request is completed when the ImageCaptureEventsMBS subclass receives a deviceDidOpenSessionWithError event. No more events will be sent to the delegate if this request fails.

7.9.27 requestSendMessage(messageCode as UInt32, data as MemoryBlock, maxReturnedDataSize as UInt64)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This method asynchronously sends an arbitrary message with optional data to a device.
Notes: This method allows developers to send a private message from a client application to a device module. This method is the functional equivalent of calling ICAObjectSendMessage() found in ImageCapture.framework, which has been deprecated in Mac OS X 10.6. The response to this command will be delivered using deviceDidSendMessage event.
The content of error returned should be examined to determine if the request completed successfully.
NOTE: This method SHOULD NOT BE USED to send PTP pass-through commands to a PTP camera. Please refer to requestSendPTPCommand defined in ICCameraDeviceMBS for sending PTP pass-through commands.

7.9.28 requestYield

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This message requests the device module in control of this device to yield control.
Notes: This message should be used only if the client is planning on communiting with the device directly. The device module may not yield control of the device if it has an open session.

7.9.29 Properties

7.9.30 AutolaunchApplicationPath as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Filesystem path of an application that is to be automatically launched when this device is added.
Notes: (Read and Write property)

7.9.31 BonjourServiceType as String

Plugin Version: 17.0, Platform: macOS, Targets: Desktop only.

Function: Service type if device was found via Bonjour..
Notes: (Read only property)

7.9.32 BskonjourServiceName as String

Plugin Version: 17.0, Platform: macOS, Targets: Desktop only.

Function: Service name if device was found via Bonjour..
Notes: (Read only property)

7.9.33 ButtonPressed as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A string object with one of the ICButtonType* values defined above.
Notes: (Read only property)

7.9.34 canDeleteAllFiles as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: If all files can be deleted.
Notes: (Read only property)

7.9.35 canDeleteOneFile as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Whether a file can be deleted.
Notes: (Read only property)

7.9.36 canEject as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Whether this device can be ejected.
Notes: (Read only property)

7.9.37 canReceiveFile as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Whether this device can receive a file.
Notes: (Read only property)

7.9.38 canSyncClock as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Whether this device can sync clock.
Notes: (Read only property)

7.9.39 canTakePicture as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Whether this device can take pictures.
Notes: (Read only property)

7.9.40 fwGUID as Int64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The FireWire GUID of a FireWire device in the IOKit registry.
Notes: This will be 0 for non-FireWire devices.
(Read only property)

7.9.41 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.9.42 HasConfigurableWiFiInterface as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the device can be configured for use on a WiFi network.
Notes: (Read only property)

7.9.43 HasOpenSession as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the device has an open session.
Notes: (Read only property)

7.9.44 Icon as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Icon image for the device.
Notes: Value is a CGImageMBS.
(Read only property)

7.9.45 IconPath as String

Plugin Version: 17.0, Platform: macOS, Targets: Desktop only.

Function: Path to icon file.
Notes: (Read only property)

7.9.46 IPAddress as String

Plugin Version: 17.0, Platform: macOS, Targets: Desktop only.

Function: IP Address.
Notes: (Read only property)

7.9.47 IsRemote as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the device is a remote device published by Image Capture device sharing facility.
Notes: (Read only property)

7.9.48 IsShared as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the device is shared using the Image Capture device sharing facility.
Notes: This value will change when sharing of this device is enabled or disabled.
(Read only property)

7.9.49 LocationDescription as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A non-localized location description string for the device.
Notes: The value returned in one of the location description strings defined above, or location obtained from the Bonjour TXT record of a network device.
(Read only property)

7.9.50 ModuleExecutableArchitecture as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Executable Architecture of the device module in control of this device.
Notes: Possible values:

I386	&h00000007
PPC	&h00000012
X86_64	&h01000007
PPC64	&h01000012

(Read only property)

7.9.51 ModulePath as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Filesystem path of the device module that is associated with this device.
Notes: Camera-specific capabilities are defined in ICCameraDeviceMBS class and scanner-specific capabilities are defined in ICScannerDeviceMBS class.
(Read only property)

7.9.52 ModuleVersion as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The bundle version of the device module associated with this device.
Notes: This may change if an existing device module associated with this device is updated or a new device module for this device is installed.
(Read only property)

7.9.53 Name as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Name of the device as reported by the device module or by the device transport when a device module is not in control of this device.
Notes: This name may change if the device module overrides the default name of the device reported by the device's transport, or if the name of the filesystem volume mounted by the device is changed by the user.
(Read only property)

7.9.54 PersistentIDString as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A string representation of the persistent ID of the device.
Notes: (Read only property)

7.9.55 ProductKind as String

Plugin Version: 17.0, Platform: macOS, Targets: Desktop only.

Function: Product kind.
Notes: (Read only property)

7.9.56 SerialNumberString as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The serial number of the device.
Notes: This will be "" if the device does not provide a serial number.
(Read only property)

7.9.57 SystemSymbolName as String

Plugin Version: 20.5, Platform: macOS, Targets: Desktop only.

Function: systemSymbolName allows to query standard system symbol used to represent the device class.
Notes: New for macOS 11.0.
Using the symbol to render an appropriate device icon will ensure proper scaling for high resolution devices.
(Read only property)

7.9.58 TransportType as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The transport type used by the device.
Notes: The possible values are: ICTransportTypeUSB, ICTransportTypeFireWire, ICTransportTypeBluetooth, ICTransportTypeTCPIP, or ICTransportTypeMassStorage.
(Read only property)

7.9.59 type as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The type of the device as defined by ICDeviceType OR'd with its ICDeviceLocationType.
Notes: The type of this device can be obtained by AND'ing the value retuned by this property with an appropriate ICDeviceTypeMask. The location type of this device can be obtained by AND'ing the value retuned by this property with an appropriate ICDeviceLocationTypeMask.
(Read only property)

7.9.60 usbLocationID as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The USB location ID of a USB device in the IOKit registry.
Notes: This will be 0 for non-USB devices.
(Read only property)

7.9.61 usbProductID as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The USB product ID of a USB device in the IOKit registry.
Notes: This will be 0 for non-USB devices.
(Read only property)

7.9.62 usbVendorID as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The USB vendor ID of a USB device in the IOKit registry.
Notes: This will be 0 for non-USB devices.
(Read only property)

7.9.63 UserData as Dictionary

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A mutable dictionary to store arbitrary key-value pairs associated with a device object.
Notes: This can be used by view objects that bind to this object to store "house-keeping" information.
In Xojo, please query dictionary, modify it and assign back to this property.

(Read and Write property)

7.9.64 UUIDString as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: A string representation of the Universally Unique ID of the device.
Notes: (Read only property)

7.9.65 Constants

Device Types

Constant	Value	Description
ICDeviceLocationTypeBluetooth	&h00000800	Device found as a paired Bluetooth device.
ICDeviceLocationTypeBonjour	&h00000400	Device found over the network by searching for Bonjour services suppor Image Capture.
ICDeviceLocationTypeLocal	&h00000100	Device found directly attached to the Macintosh via its USB or FireWir
ICDeviceLocationTypeShared	&h00000200	Device found over the network by searching for devices shared by othe intosh hosts.

Image Capture Device Location Type Mask

Constant	Value	Description
ICDeviceLocationTypeMaskBluetooth	&h00000800	Mask to detect paired Bluetooth device.
ICDeviceLocationTypeMaskBonjour	&h00000400	Mask to detect a network device that publishes a Bonjour service
ICDeviceLocationTypeMaskLocal	&h00000100	Mask to detect a local (e.g., USB or FireWire) device.
ICDeviceLocationTypeMaskRemote	&h0000FE00	Mask to detect a remote (shared, Bonjour, Bluetooth) device.
ICDeviceLocationTypeMaskShared	&h00000200	Mask to detect a device by another Macintosh host.

Image Capture Device Types

Constant	Value	Description
ICDeviceTypeCamera	&h00000001	Camera device.
ICDeviceTypeScanner	&h00000002	Scanner device.

Image Capture Device Type Mask

Constant	Value	Description
ICDeviceTypeMaskCamera	&h00000001	Mask to detect a camera device.
ICDeviceTypeMaskScanner	&h00000002	Mask to detect a scanner device.

7.10 class ICScannerBandDataMBS

7.10.1 class ICScannerBandDataMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The class for band data from scanner.
Notes: If image is too big to be transferred in one big block, it's sent in little chunks using this class.

Only uses when transfermode is ICScannerTransferModeMemoryBased.
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 19.4
- MBS Xojo Plugins, version 19.4pr7

7.10.2 Methods

7.10.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.10.4 Properties

7.10.5 bigEndian as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes if the banded image data is reported in big endian.
Notes: (Read only property)

7.10.6 bitsPerComponent as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the number of bits per component for the banded image.
Notes: (Read only property)

7.10.7 bitsPerPixel as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the number of bits per pixel for banded the image.
Notes: (Read only property)

7.10.8 bytesPerRow as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Descries how many bytes are in each image band row.
Notes: (Read only property)

7.10.9 CGImage as CGImageMBS

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

Function: Builds CGImageMBS object with the data of this band.
Notes: Returns nil if data can't be represented easily as CGImage.
This function is designed for RGB or Grayscale.
(Read only property)

7.10.10 colorSyncProfilePath as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Returns the path to the color profile matching the banded data.
Notes: (Read only property)

7.10.11 dataBuffer as Memoryblock

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The pointer to the data buffer object.
Notes: Plugin returns a copy of the data when you query this property.
(Read only property)

7.10.12 dataNumRows as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the number of rows contained in the image band.
Notes: (Read only property)

7.10.13 dataSize as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the actual data size of the image band buffer.
Notes: (Read only property)

7.10.14 dataStartRow as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the start row of the image band.
Notes: (Read only property)

7.10.15 fullImageHeight as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the full image height of the banded image.
Notes: (Read only property)

7.10.16 fullImageWidth as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes the full image width of the banded image.
Notes: (Read only property)

7.10.17 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.10.18 numComponents as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Describes how many components are contained within the banded image.
Notes: (Read only property)

7.10.19 Picture as Picture

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

Function: Builds Picture object with the data of this band.
Notes: Returns nil if data can't be represented easily as CGImage.
This function is designed for RGB or Grayscale.
(Read only property)

7.10.20 pixelDataType as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Type of pixel data that is contained in the band.
Notes: See ICScannerFunctionalUnitMBS.ICScannerPixelDataType* constants.
(Read only property)

7.11 class ICScannerDeviceMBS

7.11.1 class ICScannerDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The class for a scanner device.
Notes: ICScannerDeviceMBS is a concrete subclass of ICDeviceMBS class. ICDeviceBrowserMBS creates instances of this class. In this release, an instance of ICScannerDeviceMBS class is intended to be used by the IKScannerDeviceViewMBS object. The IKScannerDeviceView class encapsulates the complexities of setting scan parameters, performing scans and saving the result. The developer should consider using IKScannerDeviceViewMBS instead of building their own views using the ICScannerDeviceMBS object.
Subclass of the ICDeviceMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.11.2 Methods

7.11.3 availableFunctionalUnitTypes as Integer()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: An array of functional unit types available on this scanner device.
Notes: This is an array of numbers whose values are of type ICScannerFunctionalUnitType.

7.11.4 cancelScan

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Cancels the current scan operation started by sending a 'requestOverviewScan' or 'requestScan'.

7.11.5 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.11.6 ICScannerStatusRequestsOverviewScan as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Constants used for device status notifications.
Notes: A non-localized notification string to indicate that the scanner is requesting an overview scan to be performed.

7.11.7 ICScannerStatusWarmingUp as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Constants used for device status notifications.
Notes: A non-localized notification string to indicate that the scanner is warming up.

7.11.8 ICScannerStatusWarmUpDone as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Constants used for device status notifications.
Notes: A non-localized notification string to indicate that the scanner has warmed up.

7.11.9 requestOverviewScan

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Starts an overview scan on selectedFunctionalUnit.
Notes: When this request is completed, the delegate will be notified using the scannerDeviceDidCompleteOverviewScanWithError event. The content of error returned should be examined to determine if the request completed successfully.

7.11.10 requestScan

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Starts a scan on selectedFunctionalUnit.
Notes: When this request is completed, the delegate will be notified using the scannerDeviceDidCompleteScanWithError event. The content of error returned should be examined to determine if the request completed successfully.

7.11.11 requestSelectFunctionalUnit(type as Integer)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Requests the scanner device to select a functional unit.
Notes: When this request is completed, the delegate will be notified using the scannerDeviceDidSelectFunctionalUnit.

7.11.12 Properties

7.11.13 documentName as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The document name.
Notes: (Read and Write property)

7.11.14 documentUTI as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The document UTI.
Notes: Currently supported UTIs are: kUTTypeJPEG, kUTTypeJPEG2000, kUTTypeTIFF, kUTTypePNG etc.
see UTTypeMBS module.
(Read and Write property)

7.11.15 downloadsDirectory as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.11.16 downloadsFolder as FolderItem

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.11.17 maxMemoryBandSize as UInt64

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The total maximum band size requested when performing a ICScannerTransferModeMemoryBased.
Notes: (Read and Write property)

7.11.18 selectedFunctionalUnit as ICScannerFunctionalUnitMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The currently selected functional unit on the scanner device.
Notes: (Read only property)

7.11.19 transferMode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The transfer mode for scanned document.
Notes: (Read and Write property)

7.11.20 Constants

Transfer Modes

Constant	Value	Description
ICScannerTransferModeFileBased	0	Save the scan as a file.
ICScannerTransferModeMemoryBased	1	Transfer the scan as data.

7.12 class ICScannerFeatureBooleanMBS

7.12.1 class ICScannerFeatureBooleanMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFeatureBoolean object is used to represent a property of a scanner functional unit whose value can be true or false.
Notes: Subclass of the ICScannerFeatureMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.12.2 Methods

7.12.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.12.4 Properties

7.12.5 value as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The value of this feature.
Notes: (Read and Write property)

7.13 class ICScannerFeatureEnumerationMBS

7.13.1 class ICScannerFeatureEnumerationMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFeatureEnumeration object is used to represent a feature of a scanner functional unit that can have one of several discrete values.
Notes: Subclass of the ICScannerFeatureMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.13.2 Methods

7.13.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.13.4 menuItemLabels as String()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The human readable menu item labels to be used in a menu to allow the user to select the current value from an array of possible values.

7.13.5 menuItemLabelsTooltips as String()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Tooltip text associated with the menu items.

7.13.6 values as Variant()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: An array of possible values. All items in this array must be of same type.

7.13.7 Properties

7.13.8 currentValue as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The current value.
Notes: The current value can be set to one of the possible values in the ”values” property below.
(Read and Write property)

7.13.9 defaultValue as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The default value.
Notes: The default value can be set to one of the possible values in the ”values” property below.
(Read only property)

7.14 class ICScannerFeatureMBS

7.14.1 class ICScannerFeatureMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFeature class is an abstract base class used to describe a scanner feature.
Notes: ImageCaptureCore defines three concrete subclasses of ICScannerFeatureMBS: ICScannerFeatureEnumerationMBS, ICScannerFeatureRangeMBS and ICScannerFeatureBooleanMBS.
The scanner functional units may have one or more instances of these classes to allow users to choose scanner-specific settings or operations before performing a scan.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.14.2 Methods

7.14.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.14.4 Properties

7.14.5 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.14.6 humanReadableName as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The human readable name of this feature.
Notes: (Read only property)

7.14.7 internalName as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The internal name of this feature.
Notes: (Read only property)

7.14.8 tooltip as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Tooltip text describing the feature.
Notes: (Read only property)

7.14.9 type as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Scanner feature type.
Notes: See ICScannerFeatureType* constants.
(Read only property)

7.14.10 Constants

Feature Types

Constant	Value	Description
ICScannerFeatureTypeBoolean	2	The value of this feature can be true or false.
ICScannerFeatureTypeEnumeration	0	This feature can have one of several discrete values, strings or numbers.
ICScannerFeatureTypeRange	1	This value of this feature lies within a range.
ICScannerFeatureTypeTemplate	3	A group of features.

7.15 class ICScannerFeatureRangeMBS

7.15.1 class ICScannerFeatureRangeMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFeatureRange object is used to represent a property of a scanner functional unit whose value lies within a range.
Notes: Subclass of the ICScannerFeatureMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.15.2 Methods

7.15.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.15.4 Properties

7.15.5 currentValue as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The current value.
Notes: Attempting to set the current value to a value that is not coincident with a step will result in a value corresponding to the nearest step being assigned to the current value.
(Read only property)

7.15.6 defaultValue as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The default value.
Notes: Attempting to set the default value to a value that is not coincident with a step will result in a value corresponding to the nearest step being assigned to the default value.
(Read only property)

7.15.7 maxValue as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The maximum value.
Notes: (Read only property)

7.15.8 minValue as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The minimum value.
Notes: (Read only property)

7.15.9 stepSize as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The step size.
Notes: (Read only property)

7.16 class ICScannerFeatureTemplateMBS

7.16.1 class ICScannerFeatureTemplateMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFeatureTemplate object is used to define a group of one or more rectangular scan areas that can be used with a scanner functional unit.
Notes: Subclass of the ICScannerFeatureMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.16.2 Methods

7.16.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.16.4 targets as ICScannerFeatureMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The target features.

7.17 class ICScannerFunctionalUnitDocumentFeederMBS

7.17.1 class ICScannerFunctionalUnitDocumentFeederMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFunctionalUnitDocumentFeeder is a concrete subclass of ICScannerFunctionalUnit class.
Notes: ICScannerDevice creates instances of this class.
This represents the document feeder unit on the scanner.
Subclass of the ICScannerFunctionalUnitMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.17.2 Methods

7.17.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.17.4 Properties

7.17.5 documentLoaded as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the feeder has documents to scan.
Notes: This value will change when the document is loaded or removed from the feeder, if the scanner module has the capability to detect this state.
(Read only property)

7.17.6 duplexScanningEnabled as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether duplex scanning is enabled.
Notes: (Read and Write property)

7.17.7 evenPageOrientation as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Desired orientation of the even pages of the scanned document.
Notes: This property is set to ICEXIFOrientation1 initially.

Possible values:

ICEXIFOrientation1	1	Normal
ICEXIFOrientation2	2	Flipped horizontally
ICEXIFOrientation3	3	Rotated 180¬∞
ICEXIFOrientation4	4	Flipped vertically
ICEXIFOrientation5	5	Rotated 90¬∞ CCW and flipped vertically
ICEXIFOrientation6	6	Rotated 90¬∞ CCW
ICEXIFOrientation7	7	Rotated 90¬∞ CW and flipped vertically
ICEXIFOrientation8	8	Rotated 90¬∞ CW

(Read and Write property)

7.17.8 oddPageOrientation as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Desired orientation of the odd pages of the scanned document.
Notes: This property is set to ICEXIFOrientation1 initially.

Possible values:

ICEXIFOrientation1	1	Normal
ICEXIFOrientation2	2	Flipped horizontally
ICEXIFOrientation3	3	Rotated 180¬∞
ICEXIFOrientation4	4	Flipped vertically
ICEXIFOrientation5	5	Rotated 90¬∞ CCW and flipped vertically
ICEXIFOrientation6	6	Rotated 90¬∞ CCW
ICEXIFOrientation7	7	Rotated 90¬∞ CW and flipped vertically
ICEXIFOrientation8	8	Rotated 90¬∞ CW

(Read and Write property)

7.17.9 reverseFeederPageOrder as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the document feeder reads pages from back to front.
Notes: (Read only property)

7.17.10 supportsDuplexScanning as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates whether duplex scanning is supported.
Notes: (Read only property)

7.18 class ICScannerFunctionalUnitFlatbedMBS

7.18.1 class ICScannerFunctionalUnitFlatbedMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFunctionalUnitFlatbedMBS is a concrete subclass of ICScannerFunctionalUnitMBS class.
Notes: ICScannerDevice creates instances of this class.
This represents the flatbed unit on the scanner.
Subclass of the ICScannerFunctionalUnitMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.18.2 Methods

7.18.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.19 class ICScannerFunctionalUnitMBS

7.19.1 class ICScannerFunctionalUnitMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFunctionalUnit is an abstract class that represents a scanner functiona unit.
Notes: ImageCaptureCore defines three concrete subclasses of ICScannerFunctionalUnit: ICScannerFunctionalUnitFlatbed, ICScannerFunctionalUnitPositiveTransparency, ICScannerFunctionalUnitNegativeTransparency and ICScannerFunctionalUnitDocumentFeeder. ICScannerDevice creates instances of these concrete subclasses.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.19.2 Methods

7.19.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.19.4 templates as ICScannerFeatureTemplateMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: An array of objects of type ICScannerFeatureTemplate.

7.19.5 vendorFeatures as ICScannerFeatureMBS()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: An array of objects of type ICScannerFeature.

7.19.6 Properties

7.19.7 acceptsThresholdForBlackAndWhiteScanning as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if this functional unit accepts threshold value to be used when performing a scan in black & white.
Notes: (Read only property)

7.19.8 bitDepth as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The bit depth to use when performing the final scan.
Notes: This will always be one of the supported bit depths.
(Read and Write property)

7.19.9 canPerformOverviewScan as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if this functional unit can perfrom an overview scan.
Notes: Not all functional units can perform an overview scan. For example, a document feeder or a sheet feeder unit cannot perform an overview scan.
(Read only property)

7.19.10 defaultThresholdForBlackAndWhiteScanning as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Default threshold value used when performing a scan in black & white.
Notes: This value is from 0 to 255.
(Read only property)

7.19.11 documentSize as NSSizeMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Document size of the current document type expressed in current measurement unit.
Notes: (Read only property)

7.19.12 documentType as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current document type.
Notes: This will always be one of the supported document types.
(Read and Write property)

7.19.13 measurementUnit as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current measurement unit. This will always be one of the supported measurement units.
Notes: (Read and Write property)

7.19.14 nativeXResolution as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Optical resolution along the X axis.
Notes: (Read only property)

7.19.15 nativeYResolution as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Optical resolution along the Y axis.
Notes: (Read only property)

7.19.16 overviewImage as Variant

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Overview scan image.
Notes: This property will be nil for functional units that do not support overview scans.
Value is a CGImageMBS.
(Read only property)

7.19.17 overviewResolution as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Overview image resolution.
Notes: Value assigned to this will be contrained by resolutions allowed by the device.

(Read and Write property)

7.19.18 overviewScanInProgress as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if an overview scan is in progress.
Notes: (Read only property)

7.19.19 physicalSize as NSSizeMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Physical size of the scan area in current measurement unit.
Notes: (Read only property)

7.19.20 pixelDataType as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The pixel data type.
Notes: See ICScannerPixelDataType* constants.
(Read and Write property)

7.19.21 preferredResolutions as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current scan resolution.
Notes: This will always be one of the supported resolution values.
(Read only property)

7.19.22 preferredScaleFactors as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Preferred scale factors in percentage.
Notes: (Read only property)

7.19.23 resolution as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current scan resolution.
Notes: This will always be one of the supported resolution values.
(Read and Write property)

7.19.24 scaleFactor as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current scale factor.
Notes: This will always be one of the supported scale factor values.
(Read only property)

7.19.25 scanArea as NSRectMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: This property along with scanAreaOrientation describes the area to be scanned.
Notes: (Read and Write property)

7.19.26 scanAreaOrientation as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Desired orientation of the scan area.
Notes: This property along with scanArea describes the area to be scanned.
This property is set to ICEXIFOrientation1 initially. This property is not used by the ICScannerFunctionalUnitDocumentFeeder subclass.

Possible values:

(Read and Write property)

7.19.27 scanInProgress as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

ICEXIFOrientation1	1	Normal
ICEXIFOrientation2	2	Flipped horizontally
ICEXIFOrientation3	3	Rotated 180¬∞
ICEXIFOrientation4	4	Flipped vertically
ICEXIFOrientation5	5	Rotated 90¬∞ CCW and flipped vertically
ICEXIFOrientation6	6	Rotated 90¬∞ CCW
ICEXIFOrientation7	7	Rotated 90¬∞ CW and flipped vertically
ICEXIFOrientation8	8	Rotated 90¬∞ CW

Function: Indicates if a scan is in progress.
Notes: (Read only property)

7.19.28 scanProgressPercentDone as Double

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates percentage of scan completed.
Notes: (Read only property)

7.19.29 state as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates the current state of the functional unit.
Notes: See ICScannerFunctionalUnitState* constants.
(Read only property)

7.19.30 supportedBitDepths as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Supported bit depths.
Notes: The values in this set are valid values defined by ICScannerBitDepth.
(Read only property)

7.19.31 supportedDocumentTypes as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Supported document types.
Notes: The values in this set are valid values defined by ICScannerDocumentType.
(Read only property)

7.19.32 supportedMeasurementUnits as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Supported measurement units. The values in this set are valid values defined by ICScannerMeasurementUnit.
Notes: (Read only property)

7.19.33 supportedResolutions as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Supported scan resolutions in DPI.
Notes: (Read only property)

7.19.34 supportedScaleFactors as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Supported scale factors in percentage.
Notes: (Read only property)

7.19.35 thresholdForBlackAndWhiteScanning as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Threshold value to be used when performing a scan in black & white.
Notes: This value should be from 0 to 255.
(Read and Write property)

7.19.36 type as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Functional unit type.
Notes: See ICScannerFunctionalUnitType* constants.

(Read only property)

7.19.37 usesThresholdForBlackAndWhiteScanning as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if this functional unit uses threshold value to be used when performing a scan in black & white.
Notes: (Read only property)

7.19.38 Constants

Bit Depths

Constant	Value	Description
ICScannerBitDepth16Bits	16	Image with 16 bits per channel.
ICScannerBitDepth1Bit	1	1-bit image.
ICScannerBitDepth8Bits	8	Image with 8 bits per channel.

Color Data Formats

Constant	Value	Description
ICScannerColorDataFormatTypeChunky	0	For multi-channel data (e.g., RGB) data from all channels are interlea Identifies color data formats. Only relevant for multi-channel data. sponds to ”ICAP_PLANARCHUNKY” of the TWAIN Specification.
ICScannerColorDataFormatTypePlanar	1	For multi-channel data (e.g., RGB) each channel is transferred sequen Identifies color data formats. Only relevant for multi-channel data. sponds to ”ICAP_PLANARCHUNKY” of the TWAIN Specification.

Document Types

Constant	Value	Description
ICScannerDocumentType10	25	A10, 26.00 mm x 37.00 mm
ICScannerDocumentType10R	67	10R, 10” x 12” 254.00 mm x 304.80 mm 5:6
ICScannerDocumentType110	72	Instamatic 110, 13.00 mm x 17.00 mm
ICScannerDocumentType11R	69	11R, 11” x 14” 279.40 mm x 355.60 mm 11:14
ICScannerDocumentType12R	70	12R, 12” x 15” 304.80 mm x 381.00 mm 4:5
ICScannerDocumentType135	76	Standard 35 mm, 36.00 mm x 24.00 mm
ICScannerDocumentType2A0	18	2A0, 1189.00 mm x 1682.00 mm
ICScannerDocumentType3R	61	3R, 3.5” x 5” 88.90 mm x 127.00 mm 7:10
ICScannerDocumentType4A0	17	4A0, 1682.00 mm x 2378.00 mm
ICScannerDocumentType4R	62	4R, 4” x 6” 101.60 mm x 152.40 mm 2:3
ICScannerDocumentType5R	63	5R, 5” x 7” 127.00 mm x 177.80 mm 5:7
ICScannerDocumentType6R	64	6R, 6” x 8” 152.40 mm x 203.20 mm 3:4
ICScannerDocumentType8R	65	8R, 8” x 10” 203.20 mm x 254.00 mm 4:5
ICScannerDocumentTypeA0	19	A0, 841.00 mm x 1189.00 mm
ICScannerDocumentTypeA1	20	A1, 594.00 mm x 841.00 mm
ICScannerDocumentTypeA2	21	A2, 420.00 mm x 594.00 mm
ICScannerDocumentTypeA3	11	A3, 297.00 mm x 420.00 mm
ICScannerDocumentTypeA4	1	A4, 210.00 mm x 297.00 mm
ICScannerDocumentTypeA5	5	A5, 148.00 mm x 210.00 mm
ICScannerDocumentTypeA6	13	A6, 105.00 mm x 148.00 mm
ICScannerDocumentTypeA7	22	A7, 74.00 mm x 105.00 mm
ICScannerDocumentTypeA8	23	A8, 52.00 mm x 74.00 mm
ICScannerDocumentTypeA9	24	A9, 37.00 mm x 52.00 mm
ICScannerDocumentTypeAPSC	74	APS Classic, 25.10 mm x 16.70 mm
ICScannerDocumentTypeAPSH	73	APS High Definition, 30.20 mm x 16.70 mm
ICScannerDocumentTypeAPSP	75	APS Panoramic, 30.20 mm x 9.50 mm
ICScannerDocumentTypeB5	2	B5/JIS B5, 182.00 mm x 257.00 mm
ICScannerDocumentTypeBusinessCard	53	Business Card, 90.00 mm x 55.00 mm
ICScannerDocumentTypeC0	44	C0, 917.00 mm x 1297.00 mm
ICScannerDocumentTypeC1	45	C1, 648.00 mm x 917.00 mm
ICScannerDocumentTypeC10	51	C10, 28.00 mm x 40.00 mm
ICScannerDocumentTypeC2	46	C2, 458.00 mm x 648.00 mm
ICScannerDocumentTypeC3	47	C3, 324.00 mm x 458.00 mm
ICScannerDocumentTypeC4	14	C4, 229.00 mm x 324.00 mm
ICScannerDocumentTypeC5	15	C5, 162.00 mm x 229.00 mm
ICScannerDocumentTypeC6	16	C6, 114.00 mm x 162.00 mm
ICScannerDocumentTypeC7	48	C7, 81.00 mm x 114.00 mm
ICScannerDocumentTypeC8	49	C8, 57.00 mm x 81.00 mm
ICScannerDocumentTypeC9	50	C9, 40.00 mm x 57.00 mm
ICScannerDocumentTypeDefault	0	This is the platten size. Not valid for scanners without a platten.
ICScannerDocumentTypeE	60	Japanese E, 3.25” x 4.75” 82.55 mm x 120.65 mm 11:16
ICScannerDocumentTypeISOB0	26	ISO B0, 1000.00 mm x 1414.00 mm
ICScannerDocumentTypeISOB1	27	ISO B1, 707.00 mm x 1000.00 mm
ICScannerDocumentTypeISOB10	33	ISO B10, 31.00 mm x 44.00 mm
ICScannerDocumentTypeISOB2	28	ISO B2, 500.00 mm x 707.00 mm
ICScannerDocumentTypeISOB3	12	B3/ISO B3, 353.00 mm x 500.00 mm
ICScannerDocumentTypeISOB4	6	B4/ISO B4, 250.00 mm x 353.00 mm
ICScannerDocumentTypeISOB5	29	ISO B5, 176.00 mm x 250.00 mm
ICScannerDocumentTypeISOB6	7	B6/ISO B6, 125.00 mm x 176.00 mm
ICScannerDocumentTypeISOB7	30	ISO B7, 88.00 mm x 125.00 mm
ICScannerDocumentTypeISOB8	31	ISO B8, 62.00 mm x 88.00 mm
ICScannerDocumentTypeISOB9	32	ISO B9, 44.00 mm x 62.00 mm
ICScannerDocumentTypeJISB0	34	JIS B0, 1030.00 mm x 1456.00 mm
ICScannerDocumentTypeJISB1	35	JIS B1, 728.00 mm x 1030.00 mm
ICScannerDocumentTypeJISB10	43	JIS B10, 32.00 mm x 45.00 mm
ICScannerDocumentTypeJISB2	36	JIS B2, 515.00 mm x 728.00 mm
ICScannerDocumentTypeJISB3	37	JIS B3, 364.00 mm x 515.00 mm
ICScannerDocumentTypeJISB4	38	JIS B4, 257.00 mm x 364.00 mm

Scanner States

Constant	Value	Description
ICScannerFunctionalUnitStateOverviewScanInProgress	4	The scanner functional unit is performing an overview s
ICScannerFunctionalUnitStateReady	1	The scanner functional unit is ready for operation.
ICScannerFunctionalUnitStateScanInProgress	2	The scanner functional unit is performing a scan.

Scanner Functional Unit Types

Constant	Value	Description
ICScannerFunctionalUnitTypeDocumentFeeder	3	Document feeder functional unit.
ICScannerFunctionalUnitTypeFlatbed	0	Flatbed functional unit.
ICScannerFunctionalUnitTypeNegativeTransparency	2	Transparency functional unit for scanning negatives.
ICScannerFunctionalUnitTypePositiveTransparency	1	Transparency functional unit for scanning positives.

Measurement Units

Constant	Value	Description
ICScannerMeasurementUnitCentimeters	1	1 cm = 1.00 cm or 1/2.54 inches
ICScannerMeasurementUnitInches	0	1 inch = 2.54 cm
ICScannerMeasurementUnitPicas	2	1 pica = .42333333 cm or 1/6 inches
ICScannerMeasurementUnitPixels	5	
ICScannerMeasurementUnitPoints	3	1 point = .0352777775 cm or 1/72 inches
ICScannerMeasurementUnitTwips	4	1 twip = .0001763888 cm or 1/1440 inches

Pixel data types

Constant	Value	Description
ICScannerPixelDataTypeBW	0	Monochrome 1 bit pixel image.
ICScannerPixelDataTypeCIEXYZ	8	Color image in CIEXYZ color space.
ICScannerPixelDataTypeCMY	4	Color image in CMY color space.
ICScannerPixelDataTypeCMYK	5	Color image in CMYK color space.
ICScannerPixelDataTypeGray	1	8 bit pixel Gray color space.
ICScannerPixelDataTypePalette	3	Indexed Color image.
ICScannerPixelDataTypeRGB	2	Color image RGB color space.
ICScannerPixelDataTypeYUV	6	Color image in YUV color space.
ICScannerPixelDataTypeYUVK	7	Color image in YUVK color space.

7.20 class ICScannerFunctionalUnitNegativeTransparencyMBS

7.20.1 class ICScannerFunctionalUnitNegativeTransparencyMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFunctionalUnitNegativeTransparencyMBS is a concrete subclass of ICScannerFunctionalUnitMBS class.
Notes: ICScannerDeviceMBS creates instances of this class.
This represents the transparency unit on the scanner for scanning negatives.
Subclass of the ICScannerFunctionalUnitMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.20.2 Methods

7.20.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.21 class ICScannerFunctionalUnitPositiveTransparencyMBS

7.21.1 class ICScannerFunctionalUnitPositiveTransparencyMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: ICScannerFunctionalUnitPositiveTransparencyMBS is a concrete subclass of ICScannerFunctionalUnitMBS class.
Notes: ICScannerDeviceMBS creates instances of this class.
This represents the transparency unit on the scanner for scanning postives.
Subclass of the ICScannerFunctionalUnitMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

7.21.2 Methods

7.21.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

7.22 control IKCameraDeviceViewControlMBS

7.22.1 control IKCameraDeviceViewControlMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Camera Device View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.22.2 Properties

7.22.3 View as IKCameraDeviceViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The camera device view used in this control.
Notes: (Read only property)

7.22.4 Events

7.22.5 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.22.6 Close

Plugin Version: 14.3, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

7.22.7 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.

7.22.8 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.

7.22.9 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.

7.22.10 DidDownloadFile(CameraFile as ICCameraFileMBS, URL as string, File as folderItem, data as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent for each file that gets downloaded.
Notes: Based on the IKCameraDeviceViewDisplayMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.22.11 DidEncounterError(Error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the camera device reports an error.

7.22.12 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.

7.22.13 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.

7.22.14 GotFocus

Plugin Version: 16.5, 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.

7.22.15 LostFocus

Plugin Version: 16.5, 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.

7.22.16 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.

7.22.17 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.

7.22.18 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.

7.22.19 Open

Plugin Version: 14.3, 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.

7.22.20 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.

7.22.21 SelectionDidChange

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.

7.22.22 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.

7.23 class IKCameraDeviceViewMBS

7.23.1 class IKCameraDeviceViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Displays content of a Image Capture supported camera.
Notes: Subclass of the NSViewMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.23.2 Methods

7.23.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new box view with size 100/100 and position 0/0
Example:

```
dim x as new IKCameraDeviceViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 7.23.4 Constructor(Handle as Integer) 1003
- 7.23.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1004

7.23.4 Constructor(Handle as Integer)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSView handle.
Example:

```
dim t as new IKCameraDeviceViewMBS(0, 0, 100, 100)
dim v as new IKCameraDeviceViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a IKCameraDeviceView and the plugin retains this handle.
See also:

- 7.23.3 Constructor 1003
- 7.23.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1004

7.23.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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 IKCameraDeviceViewMBS(left, top, width, height)
```

Notes: On success the handle property is not zero.
See also:

- 7.23.3 Constructor 1003
- 7.23.4 Constructor(Handle as Integer) 1003

7.23.6 deleteSelectedItems

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Delete selected items.

7.23.7 downloadAllItems

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Download all items.

7.23.8 downloadSelectedItems

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Download selected items.

7.23.9 rotateLeft

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Rotate selected items left.

7.23.10 rotateRight

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Rotate selected items right.

7.23.11 selectIndexes(indexes as NSIndexSetMBS, extend as boolean)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Setting current user selection.

7.23.12 Properties

7.23.13 cameraDevice as ICCameraDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The camera device.
Notes: (Read and Write property)

7.23.14 canDeleteSelectedItems as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the user selected items can be deleted.
Notes: (Read only property)

7.23.15 canDownloadSelectedItems as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the user selected items can be downloaded.
Notes: (Read only property)

7.23.16 canRotateSelectedItemsLeft as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the user selected items can be rotated left.
Notes: (Read only property)

7.23.17 canRotateSelectedItemsRight as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Indicates if the user selected items can be rotated right.
Notes: (Read only property)

7.23.18 displaysDownloadsDirectoryControl as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Show a downloads directory control.
Notes: (Read and Write property)

7.23.19 displaysPostProcessApplicationControl as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Show a postprocessing application control.
Notes: (Read and Write property)

7.23.20 downloadAllControlLabel as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Label for the 'Download All' control - allows for example renaming to 'Import All'.
Notes: (Read and Write property)

7.23.21 downloadsDirectory as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.23.22 downloadSelectedControlLabel as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Label for the 'Download Selected' control.
Notes: (Read and Write property)

7.23.23 downloadsFolder as FolderItem

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.23.24 hasDisplayModeIcon as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Support icon view display mode.
Notes: (Read and Write property)

7.23.25 hasDisplayModeTable as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Support table view display mode.
Notes: (Read and Write property)

7.23.26 iconSize as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: in icon mode: size of the image thumbnails.
Notes: (Read and Write property)

7.23.27 mode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current display mode.
Notes: see IKCameraDeviceViewDisplayMode constants.
(Read and Write property)

7.23.28 postProcessApplication as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Postprocessing application.
Notes: A file URL to application.
(Read and Write property)

7.23.29 selectedIndexes as NSIndexSetMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current user selection.
Notes: (Read only property)

7.23.30 transferMode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Transfer mode either file based - or - in memory.
Notes: See IKCameraDeviceViewTransferMode constants.
(Read and Write property)

7.23.31 Events

7.23.32 DidDownloadFile(CameraFile as ICCameraFileMBS, URL as string, File as folderItem, data as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent for each file that gets downloaded.
Notes: Based on the IKCameraDeviceViewDisplayMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.23.33 DidEncounterError(Error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the camera device reports an error.

7.23.34 SelectionDidChange

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.

7.23.35 Constants

Display Modes

Constant	Value	Description
IKCameraDeviceViewDisplayModeIcon	1	Show Icons
IKCameraDeviceViewDisplayModeTable	0	Show Table

Transfer Modes

Constant	Value	Description
IKCameraDeviceViewTransferModeFileBased	0	File based download.
IKCameraDeviceViewTransferModeMemoryBased	1	Memory based download.

7.24 control IKDeviceBrowserViewControlMBS

7.24.1 control IKDeviceBrowserViewControlMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Device Browser View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.24.2 Properties

7.24.3 View as IKDeviceBrowserViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The device browser view used in this control.
Notes: (Read only property)

7.24.4 Events

7.24.5 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.24.6 Close

Plugin Version: 14.3, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

7.24.7 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.

7.24.8 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.

7.24.9 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.

7.24.10 DidEncounterError(error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the device browser reports an error.

7.24.11 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.

7.24.12 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.

7.24.13 GotFocus

Plugin Version: 16.5, 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.

7.24.14 LostFocus

Plugin Version: 16.5, 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.

7.24.15 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.

7.24.16 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.

7.24.17 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.

7.24.18 Open

Plugin Version: 14.3, 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.

7.24.19 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.

7.24.20 SelectionDidChange(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.
Notes: The device may be a ICCameraDeviceMBS or a ICScannerDeviceMBS.

7.24.21 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.

7.25 class IKDeviceBrowserViewMBS

7.25.1 class IKDeviceBrowserViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Displays Image Capture cameras and scanners.
Notes: Subclass of the NSViewMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.25.2 Methods

7.25.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new box view with size 100/100 and position 0/0
Example:

```
dim x as new IKDeviceBrowserViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 7.25.4 Constructor(Handle as Integer) 1015
- 7.25.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1016

7.25.4 Constructor(Handle as Integer)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSView handle.
Example:

```
dim t as new IKDeviceBrowserViewMBS(0, 0, 100, 100)
dim v as new IKDeviceBrowserViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a IKDeviceBrowserView and the plugin retains this handle.
See also:

- 7.25.3 Constructor 1015
- 7.25.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1016

7.25.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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 IKDeviceBrowserViewMBS(left, top, width, height)
```

Notes: On success the handle property is not zero.
See also:

- 7.25.3 Constructor 1015
- 7.25.4 Constructor(Handle as Integer) 1015

7.25.6 Properties

7.25.7 displaysLocalCameras as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: For device filtering - indicates that the IKDeviceBrowserView should include local cameras.
Notes: (Read and Write property)

7.25.8 displaysLocalScanners as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: for device filtering - indicates that the IKDeviceBrowserView should include local scanners.
Notes: (Read and Write property)

7.25.9 displaysNetworkCameras as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: for device filtering - indicates that the IKDeviceBrowserView should include network/shared cameras.
Notes: (Read and Write property)

7.25.10 displaysNetworkScanners as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: for device filtering - indicates that the IKDeviceBrowserView should include network/shared scanners.
Notes: (Read and Write property)

7.25.11 mode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: One of the supported display modes (table, outline, or icon mode).
Notes: (Read and Write property)

7.25.12 selectedDevice as ICDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: User selected device (ICCameraDevice or ICScannerDevice).
Notes: (Read only property)

7.25.13 Events

7.25.14 DidEncounterError(error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the device browser reports an error.

7.25.15 SelectionDidChange(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.
Notes: The device may be a ICCameraDeviceMBS or a ICScannerDeviceMBS.

7.25.16 Constants

Display Modes

Constant	Value	Description
IKDeviceBrowserViewDisplayModeIcon	2	Icon
IKDeviceBrowserViewDisplayModeOutline	1	Outline
IKDeviceBrowserViewDisplayModeTable	0	Table

7.26 control IKScannerDeviceViewControlMBS

7.26.1 control IKScannerDeviceViewControlMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The Xojo control for a Scanner Device View.
Notes: For Xojo with Cocoa target.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.26.2 Properties

7.26.3 View as IKScannerDeviceViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The scanner view used in this control.
Notes: For Xojo with Cocoa target.
(Read only property)

7.26.4 Events

7.26.5 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

7.26.6 Close

Plugin Version: 14.3, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

7.26.7 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.

7.26.8 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.

7.26.9 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.

7.26.10 DidEncounterError(error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the scanner device reports an error.

7.26.11 DidScanToBandData(data as ICScannerBandDataMBS, scanInfo as Dictionary, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For memory a based transfer this event is sent for every time an image band of data was scanned.
Notes: The 'data' parameter describes the scanned image data. Note that rotation/cropping/image adjustments are not applied yet. The 'scanInfo' parameter contains additional information (rotation angle, ...) that should be applied once the scan is completed.

7.26.12 DidScanToURL(url as String, file as FolderItem, fileData as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For file based transfer this event is sent for each image that gets scanned.
Notes: Based on the IKScannerDeviceViewTransferMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.26.13 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.

7.26.14 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.

7.26.15 GotFocus

Plugin Version: 16.5, 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.

7.26.16 LostFocus

Plugin Version: 16.5, 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.

7.26.17 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.

7.26.18 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.

7.26.19 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.

7.26.20 Open

Plugin Version: 14.3, 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.

7.26.21 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.

7.26.22 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.

7.27 class IKScannerDeviceViewMBS

7.27.1 class IKScannerDeviceViewMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: IKScannerDeviceView displays a UI to work with Image Capture supported scanners.
Notes: Subclass of the NSViewMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.3pr2

7.27.2 Methods

7.27.3 Constructor

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new box view with size 100/100 and position 0/0
Example:

```
dim x as new IKScannerDeviceViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 7.27.4 Constructor(Handle as Integer) 1024
- 7.27.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1025

7.27.4 Constructor(Handle as Integer)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSView handle.
Example:

```
dim t as new IKScannerDeviceViewMBS(0, 0, 100, 100)
dim v as new IKScannerDeviceViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a IKScannerDeviceView and the plugin retains this handle.
See also:

- 7.27.3 Constructor 1024
- 7.27.5 Constructor(left as Double, top as Double, width as Double, height as Double) 1025

7.27.5 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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 IKScannerDeviceViewMBS(left, top, width, height)
```

Notes: On success the handle property is not zero.
See also:

- 7.27.3 Constructor 1024
- 7.27.4 Constructor(Handle as Integer) 1024

7.27.6 Properties

7.27.7 displaysDownloadsDirectoryControl as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Show a downloads directory control.
Notes: (Read and Write property)

7.27.8 displaysPostProcessApplicationControl as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Show a postprocessing application control.
Notes: (Read and Write property)

7.27.9 documentName as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Document name.
Notes: (Read and Write property)

7.27.10 downloadsDirectory as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.27.11 downloadsFolder as FolderItem

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Downloads directory.
Notes: Download location can be provided as file URL with downloadsDirectory property or as folderitem with downloadsFolder property.
(Read and Write property)

7.27.12 hasDisplayModeAdvanced as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Support advanced scanning UI.
Notes: (Read and Write property)

7.27.13 hasDisplayModeSimple as Boolean

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Support a simple scanning UI.
Notes: (Read and Write property)

7.27.14 mode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Current display mode.
Notes: See IKScannerDeviceViewDisplayMode constants.
(Read and Write property)

7.27.15 overviewControlLabel as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Label for the 'Overview' control.
Notes: (Read and Write property)

7.27.16 postProcessApplication as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Postprocessing application.
Notes: (Read and Write property)

7.27.17 scanControlLabel as String

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: label for the 'Scan' control.
Notes: (Read and Write property)

7.27.18 scannerDevice as ICScannerDeviceMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: The scanner device.
Notes: (Read and Write property)

7.27.19 transferMode as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: transfer mode either file based - or - in memory.
Notes: See IKScannerDeviceViewTransferMode constants.
(Read and Write property)

7.27.20 Events

7.27.21 DidEncounterError(error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the scanner device reports an error.

7.27.22 DidScanToBandData(data as ICScannerBandDataMBS, scanInfo as Dictionary, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For memory a based transfer this event is sent for every time an image band of data was scanned.
Notes: The 'data' parameter describes the scanned image data. Note that rotation/cropping/image adjustments are not applied yet. The 'scanInfo' parameter contains additional information (rotation angle, ...) that should be applied once the scan is completed.

7.27.23 DidScanToURL(url as String, file as FolderItem, fileData as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For file based transfer this event is sent for each image that gets scanned.
Notes: Based on the IKScannerDeviceViewTransferMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.27.24 Constants

Display Modes

Constant	Value	Description
IKScannerDeviceViewDisplayModeAdvanced	1	Advanced
IKScannerDeviceViewDisplayModeSimple	0	Simple

Transfer Modes

Constant	Value	Description
IKScannerDeviceViewTransferModeFileBased	0	File based scan.
IKScannerDeviceViewTransferModeMemoryBased	1	Memory based scan.

7.28 class ImageCaptureEventsMBS

7.28.1 class ImageCaptureEventsMBS

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

Function: Central class for Image Capture events.
Notes: Whenever you have an ICA object, the plugin will register a delegate for it and dispatch all events here.
For some view classes, events are in addition dispatched to the controls.
Blog Entries

- News from the MBS Xojo Plugins Version 20.4
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.4
- MBS Xojo Plugins, version 20.4pr4
- MBS Xojo / Real Studio Plugins, version 15.4pr6

Xojo Developer Magazine

- 18.6, page 10: News

7.28.2 Properties

7.28.3 Handle as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop only.

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

7.28.4 Events

7.28.5 cameraDeviceDidAddItem(camera as ICCameraDeviceMBS, item as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an object is added to the device.
Notes: The object may be an instance of ICCameraFolder or ICCameraFile class.

7.28.6 cameraDeviceDidAddItems(camera as ICCameraDeviceMBS, items() as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an object or objects are added to the device.
Notes: Instead of receive one event per object, an array of objects is sent.
The objects may be instances of ICCameraFolder or ICCameraFile class.

7.28.7 cameraDeviceDidBecomeReadyWithCompleteContentCatalog(camera as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the camera device is done enumerating its content and is ready to receive requests.
Notes: A session must be opened on the device in order to enumerate its content and make it ready to receive requests.

7.28.8 cameraDeviceDidChangeCapability(camera as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the capability of a device changes.
Notes: This usually happens when the device module takes control or yields control of the device.

7.28.9 cameraDeviceDidCompleteDeleteFilesWithError(camera as ICCameraDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Files have been deleted.
Notes: This event is sent after the camera device completes a delete operation initiated by sending a requestDeleteFiles event to that device.

7.28.10 cameraDeviceDidDownloadFile(file as ICCameraFileMBS, error as NSErrorMBS, options as Dictionary, device as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Download of file finished.

7.28.11 cameraDeviceDidEnableAccessRestriction(camera as ICCameraDeviceMBS)

Plugin Version: 20.4, Platform: macOS, Targets: .

Function: Tells the client when an Apple device has been locked, and media is unavailable until the restriction has been removed.
Notes: For macOS 10.15 or newer.

7.28.12 cameraDeviceDidReadData(data as Memoryblock, file as ICCameraFileMBS, error as NSErrorMBS, device as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Reading file data finished.

7.28.13 cameraDeviceDidReceiveDownloadProgressForFile(file as ICCameraFileMBS, downloadedBytes as UInt64, maxBytes as UInt64)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to provide status of the download operation.

7.28.14 cameraDeviceDidReceiveMetadataForItem(camera as ICCameraDeviceMBS, item as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the metadata requested for an item on a device is available.

7.28.15 cameraDeviceDidReceivePTPEvent(camera as ICCameraDeviceMBS, eventData as MemoryBlock)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to convey a PTP event.

7.28.16 cameraDeviceDidReceiveThumbnailForItem(camera as ICCameraDeviceMBS, item as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the thumbnail requested for an item on a device is available.

7.28.17 cameraDeviceDidRemoveAccessRestriction(camera as ICCameraDeviceMBS)

Plugin Version: 20.4, Platform: macOS, Targets: .

Function: Tells the client when an Apple device has been unlocked, paired to the host, and media is available.
Notes: For macOS 10.15 or newer.

7.28.18 cameraDeviceDidRemoveItem(camera as ICCameraDeviceMBS, item as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an object is removed from the device.
Notes: The object may be an instance of ICCameraFolder or ICCameraFile class.

7.28.19 cameraDeviceDidRemoveItems(camera as ICCameraDeviceMBS, items() as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an object or objects are removed from the device.
Notes: The objects may be instances of ICCameraFolder or ICCameraFile class. This method supercedes cameraDeviceDidRemoveItem method described above.

7.28.20 cameraDeviceDidRenameItems(camera as ICCameraDeviceMBS, items() as ICCameraItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an object or objects are renamed on the device.
Notes: The objects may be instances of ICCameraFolder or ICCameraFile class.

7.28.21 cameraDeviceDidSendPTPCommand(command as Memoryblock, data as Memoryblock, response as MemoryBlock, error as NSErrorMBS, device as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Called when a requestSendPTPCommand event got a response or error.

7.28.22 cameraDeviceDidUploadFile(fileURL as string, file as FolderItem, error as NSErrorMBS, device as ICCameraDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: A file upload was completed.

7.28.23 cameraDeviceViewDidDownloadFile(cameraDeviceView as IKCameraDeviceViewMBS, CameraFile as ICCameraFileMBS, URL as string, File as folderItem, data as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent for each file that gets downloaded.
Notes: Based on the IKCameraDeviceViewDisplayMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.28.24 cameraDeviceViewDidEncounterError(cameraDeviceView as IKCameraDeviceViewMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the camera device reports an error.

7.28.25 cameraDeviceViewSelectionDidChange(cameraDeviceView as IKCameraDeviceViewMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.

7.28.26 deviceBrowserDeviceDidChangeName(browser as ICDeviceBrowserMBS, device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent if the name of a device changes.
Notes: This happens if the device module overrides the default name of the device reported by the device's transport layer, or if the name of the filesystem volume mounted by the device is changed by the user.

7.28.27 deviceBrowserDeviceDidChangeSharingState(browser as ICDeviceBrowserMBS, device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the sharing state of a device has changes.
Notes: Any Image Capture client application can choose to share the device over the network using the sharing or webSharing facility in Image Capture.

7.28.28 deviceBrowserDidAddDevice(browser as ICDeviceBrowserMBS, device as ICDeviceMBS, moreComing as boolean)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to inform that a device has been added.
Notes: If several devices are found during the initial search, then this event is sent once for each device with the value of 'moreComing' set to true in each event except the last one.

7.28.29 deviceBrowserDidEnumerateLocalDevices(browser as ICDeviceBrowserMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent after the device browser completes sending deviceBrowserDidAddDevice event for all local devices.
Notes: Detecting locally connected devices (USB and FireWire devices) is faster than detecting devices connected using a network protocol. An Image Capture client application may use this event to update its user interface to let the user know that it has completed looking for locally connected devices and then start looking for network devices.

7.28.30 deviceBrowserDidRemoveDevice(browser as ICDeviceBrowserMBS, device as ICDeviceMBS, moreGoing as boolean)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to inform that a device has been removed.
Notes: If several devices are removed at the same time, then this event is sent once for each device with the value of 'moreGoing' set to true in each event except the last one.

7.28.31 deviceBrowserRequestsSelectDevice(browser as ICDeviceBrowserMBS, device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when an event that occurred on the device may be of interest to the client application.
Notes: In Mac OS X 10.6, this event is sent when a button is pressed on a device and the current application is the target for that button press. In the case of the button-press event, if a session is open on the device, this event will not be sent, instead the deviceDidReceiveButtonPress event is sent.

7.28.32 deviceBrowserViewDidEncounterError(deviceBrowserView as IKDeviceBrowserViewMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the device browser reports an error.

7.28.33 deviceBrowserViewSelectionDidChange(deviceBrowserView as IKDeviceBrowserViewMBS, device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the user selection did change.
Notes: The device may be a ICCameraDeviceMBS or a ICScannerDeviceMBS.

7.28.34 deviceDidBecomeReady(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the device is ready to receive requests.
Notes: A camera device is ready, when it is ready to receive requests. A scanner device is ready when its

functional units are found and the default functional unit is selected for use and is ready to receive requests. The device will become ready to receive requests only after a session is opened.

7.28.35 deviceDidChangeName(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent if the name of a device changes.
Notes: This happens if the device module overrides the default name of the device reported by the device's transport layer, or if the name of the filesystem volume mounted by the device is changed by the user.

7.28.36 deviceDidChangeSharingState(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the sharing state of a device has changes.
Notes: Any Image Capture client application can choose to share the device over the network using the sharing or webSharing facility in Image Capture.

7.28.37 deviceDidCloseSessionWithError(device as ICDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when a session is closed on a device.
Notes: This event completes the process initiated by the message "requestCloseSession" sent to the device object. This event is also sent if the device module in control of the device ceases to control the device.

7.28.38 deviceDidEncounterError(device as ICDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the device delegate when a camera or scanner device encounters an error.

7.28.39 deviceDidOpenSessionWithError(device as ICDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when a session is opened on a device.
Notes: This event completes the process initiated by the requestOpenSession sent to the device object.

7.28.40 deviceDidReceiveButtonPress(device as ICDeviceMBS, buttonType as String)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the device delegate if a button is pressed on the device.
Notes: This event is sent only if a session is open on the device. The value of 'buttonType' argument is one of the ICButtonType* values defined above.

7.28.41 deviceDidReceiveCustomNotification(device as ICDeviceMBS, notification as Dictionary, data as Memoryblock)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the device delegate the device sends a custom notification 'notification' with an arbitrary byte buffer 'data'.
Notes: This event is sent only if a session is open on the device.

7.28.42 deviceDidReceiveStatusInformation(device as ICDeviceMBS, status as Dictionary)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when status information is received from a camera or a scanner.
Notes: In Mac OS X 10.6 this event is not called for camera devices. This may change in the future releases of Mac OS X.
The 'status' dictionary contains two keys, ICStatusNotificationKey and ICLocalizedStatusNotificationKey, which are defined above. If type of 'device' is ICDeviceTypeScanner, the value of ICStatusNotificationKey will be one of the values defined in ICScannerDevice.h (e.g., ICScannerStatusWarmingUp, ICScannerStatusWarmUpDone, or ICScannerStatusRequestsOverviewScan); the value of ICLocalizedStatusNotificationKey will be a localized status information string suitable for displaying to the user.

7.28.43 deviceDidRemove(device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent to the delegate to inform that a device has been removed.

7.28.44 deviceDidSendMessage(messageCode as UInt32, data as Memoryblock, error as NSErrorMBS, device as ICDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: The call to requestSendMessage was successful.

7.28.45 scannerDeviceDidBecomeAvailable(scanner as ICScannerDeviceMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when another client closes an open session on the scanner.
Notes: Scanners require exclusive access, only one client can open a session on a scanner. The scanner is available if it does not have a session opened by another client. Attempting to open a session on a scanner that already has an open session for another client will result in an error. A client that wants to open a session on a scanner as soon as it is available should implement this method and send "requestOpenSession" message to scanner object from that method.

7.28.46 scannerDeviceDidCompleteOverviewScanWithError(scanner as ICScannerDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent after the scanner device completes an overview scan.

7.28.47 scannerDeviceDidCompleteScanWithError(scanner as ICScannerDeviceMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent after the scanner device completes a scan.

7.28.48 scannerDeviceDidScanToBandData(scanner as ICScannerDeviceMBS, Data as ICScannerBandDataMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the scanner device receives the requested scan progress notification and a band of data is sent for each notification received.
Notes: In memory transfer mode, this will send a band of size that has been selected by the client via the

maxMemoryBandSize property.

7.28.49 scannerDeviceDidScanToURL(scanner as ICScannerDeviceMBS, URL as string, file as folderitem, data as MemoryBlock)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when the scanner device receives the requested scan.
Notes: If selectedFunctionalUnit is a document feeder, then this event will be sent once for each scanned page.
This event is sent when the scanner device receives the requested scan. If selectedFunctionalUnit is a document feeder, then this event will be sent once for each scanned page.

7.28.50 scannerDeviceDidSelectFunctionalUnit(scanner as ICScannerDeviceMBS, functionalUnit as Variant, Error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent when a functional unit is selected on the scanner device.
Notes: A functional unit is selected immediately after the scanner device is instantiated and in response to requestSelectFunctionalUnit method.

7.28.51 scannerDeviceViewDidEncounterError(scannerDeviceView as IKScannerDeviceViewMBS, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: This event is sent every time the scanner device reports an error.

7.28.52 scannerDeviceViewDidScanToBandData(scannerDeviceView as IKScannerDeviceViewMBS, data as ICScannerBandDataMBS, scanInfo as Dictionary, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For memory a based transfer this event is sent for every time an image band of data was scanned.
Notes: The 'data' parameter describes the scanned image data. Note that rotation/cropping/image adjustments are not applied yet. The 'scanInfo' parameter contains additional information (rotation angle, ...) that should be applied once the scan is completed.

Only called if transfermode is ICScannerTransferModeMemoryBased.

7.28.53 scannerDeviceViewDidScanToURL(scannerDeviceView as IKScannerDeviceViewMBS, url as String, file as FolderItem, fileData as MemoryBlock, error as NSErrorMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: For file based transfer this event is sent for each image that gets scanned.
Notes: Based on the IKScannerDeviceViewTransferMode the downloaded file will be saved on disk using the 'url', or returned in memory as Memoryblock.

7.28.54 Constants

Error Constants

Constant	Value	Description
ICReturnCommunicationTimedOut	-9923	Communication between different components of Image Cap
ICReturnDeleteFilesCanceled	-9942	A request to delete files was canceled.
ICReturnDeleteFilesFailed	-9941	A request to delete files failed.
ICReturnDeviceFailedToCloseSession	-9928	Failed to close a session on a specified device.
ICReturnDeviceFailedToOpenSession	-9927	Failed to open a session on a specified device.
ICReturnDeviceFailedToTakePicture	-9944	Failed to take a tethered-capture picture on a camera devic
ICReturnDeviceIsPasscodeLocked	-9943	The device is locked with a passcode. Its contents cannot b unlocked.
ICReturnDeviceSoftwareInstallationCanceled	-9948	Software installation for the device has been canceled.
ICReturnDeviceSoftwareInstallationCompleted	-9947	Software installation for the device has completed successfu
ICReturnDeviceSoftwareInstallationFailed	-9949	Software installation for the device failed.
ICReturnDeviceSoftwareIsBeingInstalled	-9946	Failed to open session because software to communicate with installed.
ICReturnDeviceSoftwareNotAvailable	-9950	Software for the device is not available from Apple.
ICReturnDeviceSoftwareNotInstalled	-9945	Failed to open session because software to communicate wit installed.
ICReturnDownloadCanceled	-9937	A download operation was canceled.
ICReturnDownloadFailed	-9934	A non-specific error occurred while downloading a file.
ICReturnFailedToCompletePassThroughCommand	-9936	Failed to complete a pass-through (e.g., PTP pass-through)
ICReturnFailedToCompleteSendMessageRequest	-9940	A request to send a event to a device failed.
ICReturnFailedToDisabeTethering	-9939	Failed to disable tethered-capture on a camera device.
ICReturnFailedToEnabeTethering	-9938	Failed to enable tethered-capture on a camera device.
ICReturnInvalidParam	-9922	An invalid parameter was found.
ICReturnReceivedUnsolicitedScannerErrorInfo	-9933	An unsolicited error information was received from a scanne
ICReturnReceivedUnsolicitedScannerStatusInfo	-9932	An unsolicited status information was received from a scann
ICReturnScannerFailedToCompleteOverviewScan	-9930	Overview scan operation failed to complete on the specified
ICReturnScannerFailedToCompleteScan	-9931	Scan operation failed to complete on the specified scanner.
ICReturnScannerFailedToSelectFunctionalUnit	-9929	Failed to select a functional unit on the specified scanner.
ICReturnScannerInUseByLocalUser	-9925	Scanner is being used by a remote user.
ICReturnScannerInUseByRemoteUser	-9926	Scanner is being used by a local user.
ICReturnScanOperationCanceled	-9924	The scan operation is canceled.
ICReturnSuccess	0	Operation successful.
ICReturnUploadFailed	-9935	A non-specific error occurred while updownloading a file.

Chapter 8

ImageKit

8.1 control DesktopIKImageBrowserViewControlMBS

8.1.1 control DesktopIKImageBrowserViewControlMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The control to wrap a IKImageBrowserViewMBS.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 16.5pr2
- MBS Xojo / Real Studio Plugins, version 14.2pr3

8.1.2 Properties

8.1.3 Scrollview as NSScrollViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The scrollview we embed the image browser view inside.
Notes: (Read only property)

8.1.4 View as IKImageBrowserViewMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: The image browser view.
Notes: (Read only property)

8.1.5 Events

8.1.6 backgroundWasRightClickedWithEvent(e as NSEventMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks the image browser view background.
Notes: event: The event that invoked the method.

This method signals that the user either right-clicked the background or left-clicked it with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.1.7 BoundsChanged

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

8.1.8 cellWasDoubleClickedAtIndex(index as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Performs custom tasks when the user double-clicks an item in the image browser view.
Notes: index: The index of the cell.

This method signals that the user double-clicked an item in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.1.9 cellWasRightClickedAtIndex(index as Integer, e as NSEventMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks an item in the image browser view.
Notes: index: The index of the cell.

event: The event that invoked the method.

This method signals that the user either right-clicked an item in the browser or left-clicked the item with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time. Available in OS X v10.5 and later.

8.1.10 concludeDragOperation(sender as NSDraggingInfoMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous performDragOperation must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's animatesToDestination property was set to true in prepareForDragOperation, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed form the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

8.1.11 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.

8.1.12 draggingEnded(sender as NSDraggingInfoMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Implement this event to be notified when a drag operation ends in some other destination.
Notes: sender: The object sending the message; use it to get details about the dragging operation.
This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

8.1.13 draggingEntered(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

Invoked when a dragged image enters the destination but only if the destination has registered for the pasteboard data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the draggingSourceOperationMask method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return NSDragOperationNone (this is the default response if the method is not implemented by the destination). A destination will still receive draggingUpdated: and draggingExited: even if NSDragOperationNone is returned by this method.

8.1.14 draggingExited(sender as NSDraggingInfoMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).
Notes: sender: The object sending the message; use it to get details about the dragging operation.

8.1.15 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns one (and only one) of the dragging operation constants described in NSDragOperation in the NS-DraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the performDragOperation method is invoked.

You typically examine the contents of the pasteboard in the draggingEntered method, where this examination is performed only once, rather than in the draggingUpdated method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

8.1.16 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.

8.1.17 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.

8.1.18 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.

8.1.19 groupAtIndex(index as Integer) as Dictionary

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Returns the group at the specified index.
Notes: index: The index of the group you want to retrieve.

Returns a dictionary that defines the group. The keys in this dictionary can be any of the following constants: IKImageBrowserGroupStyle, IKImageBrowserGroupBackgroundColorKey, IKImageBrowserGroupTitleKey, and IKImageBrowserGroupRangeKey. For more information on these constants, see IKImageBrowserView Class Reference.

This method is optional.
Available in OS X v10.5 and later.

8.1.20 itemAtIndex(index as Integer) as IKImageBrowserItemMBS

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Returns an object for the item in an image browser view that corresponds to the specified index.
Notes: index: The index of the item you want to retrieve.

Return an IKImageBrowserItem object.

Your data source must implement this method. The returned object must implement the required methods of the IKImageBrowserItem protocol.
Available in OS X v10.5 and later.

8.1.21 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.

8.1.22 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.

8.1.23 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.

8.1.24 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.

8.1.25 moveItemsAtIndexes(indexes as NSIndexSetMBS, destinationIndex as Integer) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Signals that the specified items should be moved to the specified destination.
Notes: indexes: The indexes of the items that should be reordered.
destinationIndex: The starting index of the destination the items should be moved to.

Returns true if successful; false otherwise.

This method is optional. It is invoked by the image browser view after Image Kit determines that a reordering operation should be applied. The data source should update itself by reordering its elements.
Available in OS X v10.5 and later.

8.1.26 numberOfGroups as Integer

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Returns the number of groups in an image browser view.
Notes: Return the number of groups.

This method is optional.
Available in OS X v10.5 and later.

8.1.27 numberOfItems as Integer

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Returns the number of records managed by the data source object.
Notes: Return the number of records managed by the image browser view.

Your data source must implement this method. An IKImageView object uses this method to determine how many cells it should create and display.
Available in OS X v10.5 and later.

8.1.28 performDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous prepareForDragOperation message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's animatesToDestination was set to true in prepareForDragOperation, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

8.1.29 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent draggingEntered or draggingUpdated event returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's animatesToDestination property to true in your implementation of this event.

8.1.30 removeItemsAtIndexes(indexes as NSIndexSetMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Signals that a remove operation should be applied to the specified items.
Notes: indexes: The indexes of the items that should be removed.

This method is optional. It is invoked by the image browser after Image Kit determines that a remove operation should be applied. In response, the data source should update itself by removing the specified items.
Available in OS X v10.5 and later.

8.1.31 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.

8.1.32 selectionDidChange

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Performs custom tasks when the selection changes.
Notes: This method signals that the user changes the selection in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.1.33 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Invoked when the dragging images should be changed.
Notes: sender: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

During enumerateDraggingItemsWithOptions you may set non-acceptable drag items images to nil to hide them or use the enumeration option of NSDraggingItemEnumerationClearNonenumeratedImages If there are items that you hide, then after enumeration, you need to set the numberOfValidItemsForDrop to the number of non-hidden drag items. However, if the valid item count is 0, then it is better to return NSDragOperationNone from your implementation of draggingEntered and, or draggingUpdated instead of hiding all drag items during enumeration.
Available in OS X v10.7 and later.

8.1.34 wantsPeriodicDraggingUpdates as boolean

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Asks the destination object whether it wants to receive periodic draggingUpdated events.
Notes: Returns true if the destination wants to receive periodic draggingUpdated messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic dragging-updated events even if nothing changes.

8.1.35 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.

8.1.36 writeItemsAtIndexes(indexes as NSIndexSetMBS, pasteboard as NSPasteboardMBS) as Integer

Plugin Version: 21.5, Platform: macOS, Targets: .

Function: Signals that a drag should begin.
Notes: itemIndexes: The indexes of the items that should be dragged.
pasteboard: The pasteboard to copy the items to.

Returns the number of items written to the pasteboard.

This method is optional. It is invoked after Image Kit determines that a drag should begin, but before the drag has been started.
Available in OS X v10.5 and later.

8.2 control DesktopIKImageViewControlMBS

8.2.1 control DesktopIKImageViewControlMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a IKImageView.

8.2.2 Properties

8.2.3 View as IKImageViewMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: The image view used in this control.
Notes: (Read only property)

8.2.4 Events

8.2.5 BoundsChanged

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

8.2.6 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.

8.2.7 FocusLost

Plugin Version: 22.1, 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.

8.2.8 FocusReceived

Plugin Version: 22.1, 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.

8.2.9 FrameChanged

Plugin Version: 22.1, 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.

8.2.10 MenuBarSelected

Plugin Version: 22.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In older Xojo versions, this event is named EnableMenuItems.

8.2.11 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 22.1, 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.

8.2.12 MouseDrag(x as Integer, y as Integer)

Plugin Version: 22.1, 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.

8.2.13 MouseUp(x As Integer, y As Integer)

Plugin Version: 22.1, 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.

8.2.14 ScaleFactorChanged(NewFactor as double)

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

8.2.15 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.

8.3 class IKImageBrowserCellMBS

8.3.1 class IKImageBrowserCellMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The class for an image browser cell.
Notes: The IKImageBrowserCell class is used to display a cell conforming to the IKImageBrowserItem Protocol protocol in an IKImageBrowserView.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Real Studio Plugins, version 13.1pr9

8.3.2 Methods

8.3.3 cellState as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the current cell state of the receiver.
Notes: The IKImageBrowserView creates thumbnails asynchronously. This method returns the current state.
Available in OS X v10.6 and later.

8.3.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

8.3.5 frame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's frame rectangle, which defines its position in its IKImageBrowserView.
Notes: Available in OS X v10.6 and later.

8.3.6 IKImageBrowserCellBackgroundLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the layer types you can pass to layerForType.
Notes: Layer displayed in the background.
Available in OS X v10.6 and later.

8.3.7 IKImageBrowserCellForegroundLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the layer types you can pass to layerForType.
Notes: Layer displayed in the foreground.
Available in OS X v10.6 and later.

8.3.8 IKImageBrowserCellPlaceHolderLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the layer types you can pass to layerForType.
Notes: Layer displayed as a placeholder when an image is not yet available.
Available in OS X v10.6 and later.

8.3.9 IKImageBrowserCellSelectionLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the layer types you can pass to layerForType.
Notes: Layer displayed as the selection.
Available in OS X v10.6 and later.

8.3.10 imageAlignment as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the position of the cell's image in the frame.

8.3.11 imageBrowserView as IKImageBrowserViewMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the view the receiver uses to display the cell.
Notes: Available in OS X v10.6 and later.

8.3.12 imageContainerFrame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's image container frame rectangle, which defines the position of the container of the thumbnail.
Notes: The coordinates of image container frame, in the IKImageBrowserView coordinate space.
The image frame is computed automatically from the image container frame by taking in account the image alignment and the image aspect ratio.
Subclasses can override this method to customize the position of the thumbnail container.
Available in OS X v10.6 and later.

8.3.13 imageFrame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's image frame rectangle, which defines the position of the thumbnail in its IKImageBrowserView.
Notes: Returns the coordinates of the frame, in the IKImageBrowserView coordinate space.
It is the developer's responsibility to compute the imageFrame such that it lies entirely within the cell's frame rectangle.
Subclasses can override this method to customize the position of the thumbnail.
Available in OS X v10.6 and later.

8.3.14 indexOfRepresentedItem as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the receiver's represented object in the datasource.
Notes: Available in OS X v10.6 and later.

8.3.15 isSelected as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether the cell is selected.
Notes: Returns true if the cell is selected, otherwise false.
Subclasses should not override this method.
Available in OS X v10.6 and later.

8.3.16 layerForType(type as string) as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns a layer for the specified position.
Notes: type: A string representing the layer location. See Cell Layer Positions for possible values.
Return the CALayer to display in the specified position.
Subclasses can override this method to add a Core Animation layer to the cell
Available in OS X v10.6 and later.

8.3.17 opacity as Double

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the opacity of the receiver.
Notes: Possible values are between 0.0 (transparent) and 1.0 (opaque).
Subclasses can override this method to customize the opacity of the cell.
Available in OS X v10.6 and later.

8.3.18 representedItem as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's represented object.
Notes: Subclasses should not override this method.
Available in OS X v10.6 and later.

8.3.19 selectionFrame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's selection frame rectangle, which defines the position of the selection rectangle in its IKImageBrowserView.
Notes: Subclasses can override this method to customize the position of the selection frame.
Available in OS X v10.6 and later.

8.3.20 subtitleFrame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's subtitle frame rectangle.
Notes: The coordinates of the subtitle frame, in the IKImageBrowserView coordinate space.

It is the developer's responsibility to compute the subtitleFrame such that it lies entirely within the cell's frame rectangle.
Subclasses can override this method to customize the position of the subtitle.

Available in OS X v10.6 and later.

8.3.21 titleFrame as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's title frame rectangle.
Notes: The coordinates of the title frame, in the IKImageBrowserView coordinate space.

It is the developer's responsibility to compute the titleFrame such that it lies entirely within the cell's frame rectangle.
Subclasses can override this method to customize the position of the title.

Available in OS X v10.6 and later.

8.3.22 Properties

8.3.23 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

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

8.3.24 Constants

Cell States

Constant	Value	Description
IKImageStateInvalid	1	The thumbnail is invalid. For example, an unsupported image is provided. Available in OS X v10.6 and later.
IKImageStateNoImage	0	Returned until a thumbnail has been created from the represented object. Available in OS X v10.6 and later.
IKImageStateReady	2	The receiver's represented object has been set and the cell is ready to display. Available in OS X v10.6 and later.

8.4 class IKImageBrowserItemMBS

8.4.1 class IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The class for items in image browser.
Blog Entries

- MBS Real Studio Plugins, version 13.1pr9

8.4.2 Methods

8.4.3 Constructor(imageUID as string, imageRepresentationType as string, imageRepresentation as Variant, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given values.

8.4.4 ItemWithCGImage(imageUID as string, Image as Variant, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given image.

8.4.5 ItemWithData(imageUID as string, Data as Memoryblock, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given data.

8.4.6 ItemWithFile(imageUID as string, file as folderitem, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given file.

8.4.7 ItemWithNSImage(imageUID as string, Image as NSImageMBS, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given image.

8.4.8 ItemWithPath(imageUID as string, path as string, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given path.

8.4.9 ItemWithURL(imageUID as string, URL as string, imageVersion as Integer = 1, imageTitle as string = "", imageSubtitle as string = "", isSelectable as boolean = true) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new item with given URL.

8.4.10 Properties

8.4.11 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

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

8.4.12 imageRepresentation as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the image to display.
Notes: (Read and Write computed property)

8.4.13 imageRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the representation type of the image to display.
Notes: (Read and Write computed property)

8.4.14 imageSubtitle as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the display subtitle of the image.
Notes: (Read and Write computed property)

8.4.15 imageTitle as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the display title of the image.
Notes: (Read and Write computed property)

8.4.16 imageUID as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns a unique string that identifies the data source item.
Notes: The image browser view uses this identifier to associate the data source item and its cache.
(Read and Write computed property)

8.4.17 imageVersion as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the version of the item.
Notes: The receiver can return a new version to let the image browser know that it should not use its cache for the item.
(Read and Write computed property)

8.4.18 isSelectable as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether this item is selectable.
Notes: True if the item is selectable; false otherwise.
(Read and Write computed property)

8.5 control IKImageBrowserViewControlMBS

8.5.1 control IKImageBrowserViewControlMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The control to wrap a IKImageBrowserViewMBS.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- New desktop controls
- MBS Xojo / Real Studio Plugins, version 16.5pr2
- MBS Xojo / Real Studio Plugins, version 14.2pr3

8.5.2 Properties

8.5.3 Scrollview as NSScrollViewMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The scrollview we embed the image browser view inside.
Notes: (Read only property)

8.5.4 View as IKImageBrowserViewMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The image browser view.
Notes: (Read only property)

8.5.5 Events

8.5.6 backgroundWasRightClickedWithEvent(e as NSEventMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks the image browser view background.
Notes: event: The event that invoked the method.

This method signals that the user either right-clicked the background or left-clicked it with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time.

Available in OS X v10.5 and later.

8.5.7 BoundsChanged

Plugin Version: 17.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

8.5.8 cellWasDoubleClickedAtIndex(index as Integer)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Performs custom tasks when the user double-clicks an item in the image browser view.
Notes: index: The index of the cell.

This method signals that the user double-clicked an item in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.5.9 cellWasRightClickedAtIndex(index as Integer, e as NSEventMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks an item in the image browser view.
Notes: index: The index of the cell.
event: The event that invoked the method.

This method signals that the user either right-clicked an item in the browser or left-clicked the item with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.5.10 Close

Plugin Version: 14.2, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

8.5.11 concludeDragOperation(sender as NSDraggingInfoMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous performDragOperation must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's animatesToDestination property was set to true in prepareForDragOperation, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed form the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

8.5.12 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.

8.5.13 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.

8.5.14 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.

8.5.15 draggingEnded(sender as NSDraggingInfoMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Implement this event to be notified when a drag operation ends in some other destination.
Notes: sender: The object sending the message; use it to get details about the dragging operation.
This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

8.5.16 draggingEntered(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

Invoked when a dragged image enters the destination but only if the destination has registered for the pasteboard data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the draggingSourceOperationMask method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return NSDragOperationNone (this is the default response if the method is not implemented by the destination). A destination will still receive draggingUpdated: and draggingExited: even if NSDragOperationNone is returned by this method.

8.5.17 draggingExited(sender as NSDraggingInfoMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).
Notes: sender: The object sending the message; use it to get details about the dragging operation.

8.5.18 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the performDragOperation method is invoked.

You typically examine the contents of the pasteboard in the draggingEntered method, where this examination is performed only once, rather than in the draggingUpdated method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

8.5.19 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.

8.5.20 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.

8.5.21 GotFocus

Plugin Version: 16.5, 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.

8.5.22 groupAtIndex(index as Integer) as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Returns the group at the specified index.
Notes: index: The index of the group you want to retrieve.

Returns a dictionary that defines the group. The keys in this dictionary can be any of the following constants: IKImageBrowserGroupStyle, IKImageBrowserGroupBackgroundColorKey, IKImageBrowserGroupTitleKey, and IKImageBrowserGroupRangeKey. For more information on these constants, see IKImageBrowserView Class Reference.

This method is optional.
Available in OS X v10.5 and later.

8.5.23 itemAtIndex(index as Integer) as IKImageBrowserItemMBS

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Returns an object for the item in an image browser view that corresponds to the specified index.
Notes: index: The index of the item you want to retrieve.

Return an IKImageBrowserItem object.

Your data source must implement this method. The returned object must implement the required methods of the IKImageBrowserItem protocol.
Available in OS X v10.5 and later.

8.5.24 LostFocus

Plugin Version: 16.5, 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.

8.5.25 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.

8.5.26 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.

8.5.27 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.

8.5.28 moveItemsAtIndexes(indexes as NSIndexSetMBS, destinationIndex as Integer) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Signals that the specified items should be moved to the specified destination.
Notes: indexes: The indexes of the items that should be reordered.
destinationIndex: The starting index of the destination the items should be moved to.

Returns true if successful; false otherwise.

This method is optional. It is invoked by the image browser view after Image Kit determines that a reordering operation should be applied. The data source should update itself by reordering its elements.
Available in OS X v10.5 and later.

8.5.29 numberOfGroups as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Returns the number of groups in an image browser view.
Notes: Return the number of groups.

This method is optional.
Available in OS X v10.5 and later.

8.5.30 numberOfItems as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Returns the number of records managed by the data source object.
Notes: Return the number of records managed by the image browser view.

Your data source must implement this method. An IKImageView object uses this method to determine how many cells it should create and display.
Available in OS X v10.5 and later.

8.5.31 Open

Plugin Version: 14.2, 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.

8.5.32 performDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked after the released image has been removed from the screen, signaling the receiver to import the pasteboard data.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous prepareForDragOperation message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's animatesToDestination was set to true in prepareForDragOperation, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

8.5.33 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent draggingEntered or draggingUpdated event returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's animatesToDestination property to true in your implementation of this event.

8.5.34 removeItemsAtIndexes(indexes as NSIndexSetMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Signals that a remove operation should be applied to the specified items.
Notes: indexes: The indexes of the items that should be removed.

This method is optional. It is invoked by the image browser after Image Kit determines that a remove operation should be applied. In response, the data source should update itself by removing the specified items.
Available in OS X v10.5 and later.

8.5.35 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.

8.5.36 selectionDidChange

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Performs custom tasks when the selection changes.
Notes: This method signals that the user changes the selection in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.5.37 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Invoked when the dragging images should be changed.
Notes: sender: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

During enumerateDraggingItemsWithOptions you may set non-acceptable drag items images to nil to hide them or use the enumeration option of NSDraggingItemEnumerationClearNonenumeratedImages If there are items that you hide, then after enumeration, you need to set the numberOfValidItemsForDrop to the number of non-hidden drag items. However, if the valid item count is 0, then it is better to return NSDragOperationNone from your implementation of draggingEntered and, or draggingUpdated instead of hiding all drag items during enumeration.
Available in OS X v10.7 and later.

8.5.38 wantsPeriodicDraggingUpdates as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Asks the destination object whether it wants to receive periodic draggingUpdated events.
Notes: Returns true if the destination wants to receive periodic draggingUpdated messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic dragging-updated events even if nothing changes.

8.5.39 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.

8.5.40 writeItemsAtIndexes(indexes as NSIndexSetMBS, pasteboard as NSPasteboardMBS) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Signals that a drag should begin.
Notes: itemIndexes: The indexes of the items that should be dragged.
pasteboard: The pasteboard to copy the items to.

Returns the number of items written to the pasteboard.

This method is optional. It is invoked after Image Kit determines that a drag should begin, but before the drag has been started.
Available in OS X v10.5 and later.

8.6 class IKImageBrowserViewMBS

8.6.1 class IKImageBrowserViewMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The IKImageBrowserView class is a view for displaying and browsing a large amount of images and movies efficiently.
Notes: Available in OS X v10.5 and later.
Subclass of the NSViewMBS class.
Blog Entries

- MBS Real Studio Plugins, version 13.1pr14
- MBS Real Studio Plugins, version 13.1pr9

8.6.2 Methods

8.6.3 cellForItemAtIndex(index as Integer) as IKImageBrowserCellMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the browser cell for the item at the specified index.
Notes: Subclasses must not override this method.
Available in OS X v10.6 and later.

8.6.4 collapseGroupAtIndex(index as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Collapses a group at the specified index.
Notes: index: The index of the group you want to collapse.

8.6.5 columnIndexesInRect(rect as NSRectMBS) as NSIndexSetMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the column indexes in the specified rectangle.
Notes: rect: The rectangle in the view's coordinate system.
Returns an index set containing the cell indexes.
Available in OS X v10.6 and later.

8.6.6 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new control with size 100/100 and position 0/0
Example:

```
dim t as new IKImageBrowserViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 8.6.7 Constructor(Handle as Integer) 1079
- 8.6.8 Constructor(left as Double, top as Double, width as Double, height as Double) 1079

8.6.7 Constructor(Handle as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given IKImageBrowserView handle.
Example:

```
dim t as new IKImageBrowserViewMBS(0, 0, 100, 100)
dim v as new IKImageBrowserViewMBS(t.handle)

MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a IKImageBrowserView and the plugin retains this handle.
See also:

- 8.6.6 Constructor 1079
- 8.6.8 Constructor(left as Double, top as Double, width as Double, height as Double) 1079

8.6.8 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new control with the given size and position.
Example:

```
dim x as new IKImageBrowserViewMBS(0, 0, 100, 20)
```

Notes: On success the handle property is not zero.
See also:

- 8.6.6 Constructor 1079
- 8.6.7 Constructor(Handle as Integer) 1079

8.6.9 Destructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The destructor.

8.6.10 dropOperation as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the current drop operation.
Notes: Returns IKImageBrowserDropOn if the drop occurs on an item, otherwise IKImageBrowserDropBefore.
The returned value is valid when a drop occurred and until next drop.
For example, given a browser with N cells , a cell of N-1 and operation of IKImageBrowserDropOn would specify a drop on the last cell. To specify a drop after the last cell, one would use an index of N and IKImageBrowserDropBefore for the operation.
Available in OS X v10.6 and later.

8.6.11 expandGroupAtIndex(index as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Expands a group at the specified index.
Notes: index: The index of the group you want to expand.
Available in OS X v10.5 and later.

8.6.12 getValue(name as String) as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Queries a value for a given key.

8.6.13 IKImageBrowserBackgroundColorKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Notes: A key for the background color of the image browser view. The associated value is a NSColorMBS object.
Available in OS X v10.5 and later.

8.6.14 IKImageBrowserCellsHighlightedTitleAttributesKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Notes: A key for the highlighted title attribute for an item in the image browser view. The associated value is a Dictionary.
Available in OS X v10.5 and later.

8.6.15 IKImageBrowserCellsOutlineColorKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Notes: A key for the outline color for an item in the image browser view. The associated value is an NSColorMBS object.
Available in OS X v10.5 and later.

8.6.16 IKImageBrowserCellsSubtitleAttributesKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Notes: A key for a subtitle attribute for an item in the image browser view. The associated value is a dictionary.
Available in OS X v10.5 and later.

8.6.17 IKImageBrowserCellsTitleAttributesKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Example:

```
dim Imagebrowser as IKImageBrowserViewMBS // your control
dim d as new Dictionary
d.Value(NSAttributedStringMBS.NSForegroundColorAttributeName) = NSColorMBS.redColor

Imagebrowser.setValue Imagebrowser.IKImageBrowserCellsTitleAttributesKey, d
```

Notes: A key for title attribute of an item in the image browser view. The associated value is a dDictionary. Available in OS X v10.5 and later.

8.6.18 IKImageBrowserCGImageRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A CGImageRef object.

8.6.19 IKImageBrowserCGImageSourceRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A CGImageSourceRef object.

8.6.20 IKImageBrowserGroupBackgroundColorKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the background color of a group. The associated value is an NSColor object. This color is used only for the bezel style.
Available in OS X v10.5 and later.

8.6.21 IKImageBrowserGroupFooterLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the header layer of the group. The associated value is a CALayer.
Available in OS X v10.6 and later.

8.6.22 IKImageBrowserGroupHeaderLayer as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the header layer of the group. The associated value is a CALayer.
Available in OS X v10.6 and later.

8.6.23 IKImageBrowserGroupRangeKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the range of a group. The associated value is a NSRangeMBS. This is required if the view uses grouping
Available in OS X v10.5 and later.

8.6.24 IKImageBrowserGroupStyleKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the style of a group. The associated value is one of the constants defined in "Group Style Attributes".
Available in OS X v10.5 and later.

8.6.25 IKImageBrowserGroupTitleKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the group attribute keys.
Notes: A key for the title of a group. The associated value is a string. This string is used for the disclosure style only.
Available in OS X v10.5 and later.

8.6.26 IKImageBrowserIconRefPathRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A path to an icon.

8.6.27 IKImageBrowserIconRefRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: An icon.

8.6.28 IKImageBrowserNSBitmapImageRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: An NSBitmapImageRep object.

8.6.29 IKImageBrowserNSDataRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: Value for this key is a memoryblock.

8.6.30 IKImageBrowserNSImageRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: An NSImage object.

8.6.31 IKImageBrowserNSURLRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: An NSURL object.

8.6.32 IKImageBrowserPathRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A path representation (string).

8.6.33 IKImageBrowserPDFPageRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A PDFPage instance or a CGPDFPageRef.

8.6.34 IKImageBrowserQCCompositionPathRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A path (String) or URL (NSURL) to a Quartz Composer composition.

8.6.35 IKImageBrowserQCCompositionRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A QCComposition object.

8.6.36 IKImageBrowserQTMoviePathRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A path (string) or URL to a QuickTime movie.

8.6.37 IKImageBrowserQTMovieRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A QTMovie object.

8.6.38 IKImageBrowserQuickLookPathRepresentationType as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the image representation types.
Notes: A path (string) or URL (NSURL) to load data using QuickLook.

8.6.39 IKImageBrowserSelectionColorKey as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: One of the keys for image browser view options.
Notes: A key for the color that indicates a selection. The associated value is an NSColorMBS object.
Available in OS X v10.5 and later.

8.6.40 indexAtLocationOfDroppedItem as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the cell where the drop operation occurred.
Notes: Returns the index of the cell where the drop operation occurred.
The returned index is valid until the next drop occurs.
Available in OS X v10.5 and later.

8.6.41 indexOfItemAtPoint(point as NSPointMBS) as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the item at the specified location.
Notes: Returns the index of the item or NSNotFound (-1) if no item at this location.

8.6.42 isGroupExpandedAtIndex(index as Integer) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether the group at the provided index is expanded.
Notes: Return true if the group is expanded; false otherwise.

8.6.43 itemFrameAtIndex(index as Integer) as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the frame rectangle for the item located at the specified index.
Notes: index: The index of the item whose frame rectangle you want to obtain.
Return the frame rectangle of the item.

8.6.44 newCellForRepresentedItem(item as IKImageBrowserItemMBS) as IKImageBrowserCellMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the cell to use for the specified item.
Notes: Subclasses can override this method to customize the appearance of the cell that will represent anItem.
Available in OS X v10.6 and later.

8.6.45 numberOfColumns as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the current number of columns.
Notes: Available in OS X v10.6 and later.

8.6.46 numberOfRows as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the current number of rows.
Notes: Available in OS X v10.6 and later.

8.6.47 rectOfColumn(columnIndex as Integer) as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle containing the specified column.
Notes: Return a rectangle containing the column. Specified in the view’s coordinate system.
Available in OS X v10.6 and later.

8.6.48 rectOfRow(rowIndex as Integer) as NSRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the rectangle containing the specified row.
Notes: Returns a rectangle containing the column. Specified in the view's coordinate system.
Available in OS X v10.6 and later.

8.6.49 reloadData

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Marks the receiver as needing its data reloaded.

8.6.50 rowIndexesInRect(rect as NSRectMBS) as NSIndexSetMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the row indexes in the specified rectangle.
Notes: rect: A rectangle in the view's coordinate system.
Returns an index set containing the item indexes.
Available in OS X v10.6 and later.

8.6.51 scrollIndexToVisible(index as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Scrolls the receiver to the item at the specified index.

8.6.52 selectionIndexes as NSIndexSetMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the indexes of the selected cells.

8.6.53 setDropIndex(index as Integer, operation as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Allows the class to retarget the drop action.
Notes: index: The requested drop index.
operation: The requested drop operation. The possible values are described in IKImageBrowserDropOperation.

For example, To specify a drop on the second item, one would specify index as 1, and operation as IKImageBrowserDropOn. To specify a drop after the last item, one would specify index as the number of items and operation as IKImageBrowserDropBefore.

Passing a value of –1 for index, and IKImageBrowserDropOn as the operation causes the entire browser view to be highlighted rather than a specific item. This is useful if the data displayed by the receiver does not allow the user to drop items at a specific item location.

8.6.54 setSelectionIndexes(indexes as NSIndexSetMBS, extendSelection as boolean = false)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Selects cells at the specified indexes.
Notes: indexes: The indexes of the cells you want to select.
extendSelection: A boolean value that specifies whether to extend the current selection. Pass true to extends the selection; false replaces the current selection.
Available in OS X v10.5 and later.

8.6.55 setValue(name as String, value as Variant)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Sets a value for a given key.

8.6.56 visibleItemIndexes as NSIndexSetMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the indexes of the view's currently visible items.
Notes: Available in OS X v10.6 and later.

8.6.57 Properties

8.6.58 allowsDroppingOnItems as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the user can drop on items.
Notes: True if the user is able to drop on items, otherwise false.
Available in OS X v10.6 and later.
(Read and Write computed property)

8.6.59 allowsEmptySelection as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether an empty selection is allowed.
Notes: (Read and Write computed property)

8.6.60 allowsMultipleSelection as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the user can select more than one cell at a time.
Notes: (Read and Write computed property)

8.6.61 allowsReordering as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the user can reorder items.
Notes: (Read and Write computed property)

8.6.62 animates as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver animates reordering and changes of the data source.
Notes: (Read and Write computed property)

8.6.63 backgroundLayer as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The Core Animation layer used as the view's background.
Notes: The background layer can have sublayers. Additionally, the layers can also contain animations.
The layer is optional.
Available in OS X v10.6 and later.
(Read and Write computed property)

8.6.64 canControlQuickLookPanel as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the view can automatically take control of the QuickLook panel.
Notes: When the browser view displays the QuickLook panel it sets itself as the QuickLook datasource. If the browser cells returned by the datasource return items that are URLs or paths, then the QuickLook panel will display the image at that location. Otherwise, the browser cell must implement the QLPreviewItem protocol and return the requested URL for the custom cell.
Available in OS X v10.6 and later.
(Read and Write computed property)

8.6.65 cellSize as NSSizeMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The cell size.
Notes: You must use CellSize or ZoomValue, but not both. Setting the zoom value changes the cell size, and vice versa.
Available in OS X v10.5 and later.
(Read and Write computed property)

8.6.66 cellsStyleMask as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The appearance style of the cells.
Notes: (Read and Write computed property)

8.6.67 constrainsToOriginalSize as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver constrains the cell's image to its original size.
Notes: The default value is false.
(Read and Write computed property)

8.6.68 contentResizingMask as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The content resizing mask, which determines how its content is resized while zooming.
Notes: You specify a mask by combining any of the following options using the bitwise OR operator: NSViewWidthSizable (2), NSViewHeightSizable (16). Other values are ignored.
(Read and Write computed property)

8.6.69 foregroundLayer as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the foreground Core Animation layer
Notes: Returns a CALayer instance.
Available in OS X v10.6 and later.
(Read and Write computed property)

8.6.70 intercellSpacing as NSSizeMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the spacing between cells in the view.
Notes: Returns the vertical and horizontal spacing between cells.
Available in OS X v10.6 and later.
(Read and Write computed property)

8.6.71 zoomValue as Double

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The zoom value.
Notes: The zoom value. This value should be greater or equal to zero and less or equal than one. A zoom value of zero corresponds to the minimum size (40x40 pixels). A zoom value of one means images fits the

browser bounds. Other values are interpolated.
Discussion
You must use ZoomValue or CellSize, but not both. Setting the zoom value changes the cell size, and vice versa.
Available in OS X v10.5 and later.
(Read and Write computed property)

8.6.72 Events

8.6.73 backgroundWasRightClickedWithEvent(e as NSEventMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks the image browser view background.
Notes: event: The event that invoked the method.

This method signals that the user either right-clicked the background or left-clicked it with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.6.74 cellWasDoubleClickedAtIndex(index as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Performs custom tasks when the user double-clicks an item in the image browser view.
Notes: index: The index of the cell.

This method signals that the user double-clicked an item in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.6.75 cellWasRightClickedAtIndex(index as Integer, e as NSEventMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Performs custom tasks when the user right-clicks an item in the image browser view.
Notes: index: The index of the cell.
event: The event that invoked the method.

This method signals that the user either right-clicked an item in the browser or left-clicked the item with the Alt key pressed. You can implement this method if you want to perform custom tasks at that time.

Available in OS X v10.5 and later.

8.6.76 concludeDragOperation(sender as NSDraggingInfoMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked when the dragging operation is complete, signaling the receiver to perform any necessary clean-up.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

For this method to be invoked, the previous performDragOperation must have returned true.

The destination implements this method to perform any tidying up that it needs to do, such as updating its visual representation now that it has incorporated the dragged data. This message is the last message sent from sender to the destination during a dragging session.

If the sender object's animatesToDestination property was set to true in prepareForDragOperation, then the drag image is still visible. At this point you should draw the final visual representation in the view. When this method returns, the drag image is removed form the screen. If your final visual representation matches the visual representation in the drag, this is a seamless transition.

8.6.77 draggingEnded(sender as NSDraggingInfoMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Implement this event to be notified when a drag operation ends in some other destination.
Notes: sender: The object sending the message; use it to get details about the dragging operation.
This method might be used by a destination doing auto-expansion in order to collapse any auto-expands.

8.6.78 draggingEntered(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked when the dragged image enters destination bounds or frame; delegate returns dragging operation to perform.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

Invoked when a dragged image enters the destination but only if the destination has registered for the pasteboard data type involved in the drag operation. Specifically, this method is invoked when the mouse pointer enters the destination's bounds rectangle (if it is a view object) or its frame rectangle (if it is a window object).

This method must return a value that indicates which dragging operation the destination will perform when the image is released. In deciding which dragging operation to return, the method should evaluate the overlap between both the dragging operations allowed by the source (obtained from sender with the draggingSourceOperationMask method) and the dragging operations and pasteboard data types the destination itself supports.

If none of the operations is appropriate, this method should return NSDragOperationNone (this is the default response if the method is not implemented by the destination). A destination will still receive draggingUpdated: and draggingExited: even if NSDragOperationNone is returned by this method.

8.6.79 draggingExited(sender as NSDraggingInfoMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked when the dragged image exits the destination's bounds rectangle (in the case of a view object) or its frame rectangle (in the case of a window object).
Notes: sender: The object sending the message; use it to get details about the dragging operation.

8.6.80 draggingUpdated(sender as NSDraggingInfoMBS) as Integer

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked periodically as the image is held within the destination area, allowing modification of the dragging operation or mouse-pointer position.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns one (and only one) of the dragging operation constants described in NSDragOperation in the NSDraggingInfo reference. The default return value (if this method is not implemented by the destination) is the value returned by the previous draggingEntered: message.

For this to be invoked, the destination must have registered for the pasteboard data type involved in the drag operation. The messages continue until the image is either released or dragged out of the window or view.

This method provides the destination with an opportunity to modify the dragging operation depending on the position of the mouse pointer inside of the destination view or window object. For example, you may have several graphics or areas of text contained within the same view and wish to tailor the dragging operation, or to ignore the drag event completely, depending upon which object is underneath the mouse pointer at the time when the user releases the dragged image and the performDragOperation method is invoked.

You typically examine the contents of the pasteboard in the draggingEntered method, where this examination is performed only once, rather than in the draggingUpdated method, which is invoked multiple times.

Only one destination at a time receives a sequence of draggingUpdated messages. If the mouse pointer is within the bounds of two overlapping views that are both valid destinations, the uppermost view receives these messages until the image is either released or dragged out.

8.6.81 groupAtIndex(index as Integer) as Dictionary

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Returns the group at the specified index.
Notes: index: The index of the group you want to retrieve.

Returns a dictionary that defines the group. The keys in this dictionary can be any of the following constants: IKImageBrowserGroupStyle, IKImageBrowserGroupBackgroundColorKey, IKImageBrowserGroupTitleKey, and IKImageBrowserGroupRangeKey. For more information on these constants, see IKImageBrowserView Class Reference.

This method is optional.
Available in OS X v10.5 and later.

8.6.82 itemAtIndex(index as Integer) as IKImageBrowserItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Returns an object for the item in an image browser view that corresponds to the specified index.
Notes: index: The index of the item you want to retrieve.

Return an IKImageBrowserItem object.

Your data source must implement this method. The returned object must implement the required methods of the IKImageBrowserItem protocol.
Available in OS X v10.5 and later.

8.6.83 moveItemsAtIndexes(indexes as NSIndexSetMBS, destinationIndex as Integer) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Signals that the specified items should be moved to the specified destination.
Notes: indexes: The indexes of the items that should be reordered.
destinationIndex: The starting index of the destination the items should be moved to.

Returns true if successful; false otherwise.

This method is optional. It is invoked by the image browser view after Image Kit determines that a reordering operation should be applied. The data source should update itself by reordering its elements.
Available in OS X v10.5 and later.

8.6.84 numberOfGroups as Integer

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Returns the number of groups in an image browser view.
Notes: Return the number of groups.

This method is optional.
Available in OS X v10.5 and later.

8.6.85 numberOfItems as Integer

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Returns the number of records managed by the data source object.
Notes: Return the number of records managed by the image browser view.

Your data source must implement this method. An IKImageView object uses this method to determine how many cells it should create and display.
Available in OS X v10.5 and later.

8.6.86 performDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked after the released image has been removed from the screen, signaling the receiver to

import the pasteboard data.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Returns if the destination accepts the data, it returns true; otherwise it returns false. The default is to return false.

For this method to be invoked, the previous prepareForDragOperation message must have returned true. The destination should implement this method to do the real work of importing the pasteboard data represented by the image.

If the sender object's animatesToDestination was set to true in prepareForDragOperation, then setup any animation to arrange space for the drag items to animate to. Also at this time, enumerate through the dragging items to set their destination frames and destination images.

8.6.87 prepareForDragOperation(sender as NSDraggingInfoMBS) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked when the image is released, allowing the receiver to agree to or refuse drag operation.
Notes: sender: The object sending the message; use it to get details about the dragging operation.

Return true if the receiver agrees to perform the drag operation and false if not.

This method is invoked only if the most recent draggingEntered or draggingUpdated event returned an acceptable drag-operation value.

If you want the drag items to animate from their current location on screen to their final location in your view, set the sender object's animatesToDestination property to true in your implementation of this event.

8.6.88 removeItemsAtIndexes(indexes as NSIndexSetMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Signals that a remove operation should be applied to the specified items.
Notes: indexes: The indexes of the items that should be removed.

This method is optional. It is invoked by the image browser after Image Kit determines that a remove operation should be applied. In response, the data source should update itself by removing the specified items.
Available in OS X v10.5 and later.

8.6.89 selectionDidChange

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Performs custom tasks when the selection changes.
Notes: This method signals that the user changes the selection in the image browser view. You can implement this method if you want to perform custom tasks at that time.
Available in OS X v10.5 and later.

8.6.90 updateDraggingItemsForDrag(sender as NSDraggingInfoMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Invoked when the dragging images should be changed.
Notes: sender: The object sending the message; use this object to get details about the dragging operation.

While a destination may change the dragging images at any time, it is recommended to wait until this method is called before updating the dragging images.

This allows the system to delay changing the dragging images until it is likely that the user will drop on this destination. Otherwise, the dragging images will change too often during the drag which would be distracting to the user.

During enumerateDraggingItemsWithOptions you may set non-acceptable drag items images to nil to hide them or use the enumeration option of NSDraggingItemEnumerationClearNonenumeratedImages If there are items that you hide, then after enumeration, you need to set the numberOfValidItemsForDrop to the number of non-hidden drag items. However, if the valid item count is 0, then it is better to return NSDragOperationNone from your implementation of draggingEntered and, or draggingUpdated instead of hiding all drag items during enumeration.
Available in OS X v10.7 and later.

8.6.91 wantsPeriodicDraggingUpdates as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Asks the destination object whether it wants to receive periodic draggingUpdated events.
Notes: Returns true if the destination wants to receive periodic draggingUpdated messages, false otherwise.

If the destination returns false, these messages are sent only when the mouse moves or a modifier flag changes. Otherwise the destination gets the default behavior, where it receives periodic dragging-updated events even if nothing changes.

8.6.92 writeItemsAtIndexes(indexes as NSIndexSetMBS, pasteboard as NSPasteboardMBS) as Integer

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Signals that a drag should begin.
Notes: itemIndexes: The indexes of the items that should be dragged.
pasteboard: The pasteboard to copy the items to.

Returns the number of items written to the pasteboard.

This method is optional. It is invoked after Image Kit determines that a drag should begin, but before the drag has been started.
Available in OS X v10.5 and later.

8.6.93 Constants

Constants

Constant	Value	Description
IKImageBrowserDropBefore	1	One of the constants to specify the locations for dropping items onto the browser view. Drop the item before the cell. Available in OS X v10.6 and later. Used by the method setDropIndex.
IKImageBrowserDropOn	0	One of the constants to specify the locations for dropping items onto the browser view. Drop the item on the cell. Available in OS X v10.6 and later. Used by the method setDropIndex.

Cell Styles

Constant	Value	Description
IKCellsStyleNone	0	No style.
IKCellsStyleOutlined	2	Cells are outlined.
IKCellsStyleShadowed	1	Cells use shadows.
IKCellsStyleSubtitled	8	Cells display a subtitle.
IKCellsStyleTitled	4	Cells display a title.

Bevel Styles

Constant	Value	Description
IKGroupBezelStyle	0	A bezel style. Available in OS X v10.5 and later.
IKGroupDisclosureStyle	1	A disclosure triangle. Available in OS X v10.5 and later.

8.7 class IKImageEditPanelMBS

8.7.1 class IKImageEditPanelMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: The class for the image edit panel from Mac OS X 10.5.
Notes: Subclass of the NSPanelMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.5pr8
- MonkeyBread Software Releases the MBS Plugins 8.1

8.7.2 Methods

8.7.3 Constructor

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new image edit panel.

8.7.4 reloadData

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Requests the panel to reload the image.
Notes: Do call this if you have a new image to return in the image event.

8.7.5 Properties

8.7.6 LastImage as Picture

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: The last image passed to you or requested from you.
Notes: This property is set with the picture you return with the image event and is set with the image sent to you useing the Changed event.
(Read and Write property)

8.7.7 Events

8.7.8 Changed(pic as picture, CGImageHandle as Integer, metaData as dictionary)

Plugin Version: 8.1, Platform: macOS, Targets: .

Function: The image changed and you should update your GUI.
Notes: pic: The image as a picture.
CGImageHandle: The internal handle to the original CGImage which is used to make the picture.
metaData: additional image data.

8.7.9 hasAdjustMode as Boolean

Plugin Version: 13.5, Platform: macOS, Targets: .

Function: Returns whether the adjust mode view tab should be displayed.
Notes: Return true if the tab should be displayed, otherwise false.
Available on Mac OS X 10.6 or newer.

8.7.10 hasDetailsMode as Boolean

Plugin Version: 13.5, Platform: macOS, Targets: .

Function: Returns whether the details mode view tab should be displayed.
Notes: True if the tab should be displayed, otherwise false.
Available on Mac OS X 10.6 or newer.

8.7.11 hasEffectsMode as Boolean

Plugin Version: 13.5, Platform: macOS, Targets: .

Function: Returns whether the effects mode view tab should be displayed.
Notes: True if the tab should be displayed, otherwise false.
Available on Mac OS X 10.6 or newer.

8.7.12 Image as picture

Plugin Version: 8.1, Platform: macOS, Targets: .

Function: The panel needs an image to start with.
Notes: Return your image in this event whenever the panel needs it.

8.7.13 imageProperties as Dictionary

Plugin Version: 13.5, Platform: macOS, Targets: .

Function: Returns a dictionary of the image properties associated with the image in the image edit panel.
Notes: Available on Mac OS X 10.5 or newer.

8.7.14 thumbnailWithMaximumSize(Width as Double, Height as Double) as picture

Plugin Version: 13.5, Platform: macOS, Targets: .

Function: Returns a thumbnail image whose size is no larger than the specified size.
Notes: Available in OS X v 10.5 and later.

8.8 control IKImageViewControlMBS

8.8.1 control IKImageViewControlMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: The Xojo control to host a IKImageView.

8.8.2 Properties

8.8.3 View as IKImageViewMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: The image view used in this control.
Notes: (Read only property)

8.8.4 Events

8.8.5 BoundsChanged

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: The event called when the bounds, but not the frame, changed.

8.8.6 Close

Plugin Version: 22.1, Platform: macOS, Targets: .

Function:

The control is about to close.
In Xojo version 2021r3 and newer this event is named Closing.

8.8.7 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.

8.8.8 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 22.1, 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.

8.8.9 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.

8.8.10 EnableMenuItems

Plugin Version: 22.1, Platform: macOS, Targets: .

Function:

The event where you can enable menu items.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

8.8.11 FrameChanged

Plugin Version: 22.1, 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.

8.8.12 GotFocus

Plugin Version: 22.1, 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.

8.8.13 LostFocus

Plugin Version: 22.1, 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.

8.8.14 MouseDown(x as Integer, y as Integer, Modifiers as Integer) As Boolean

Plugin Version: 22.1, 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.

8.8.15 MouseDrag(x as Integer, y as Integer)

Plugin Version: 22.1, 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.

8.8.16 MouseUp(x As Integer, y As Integer)

Plugin Version: 22.1, 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.

8.8.17 Open

Plugin Version: 22.1, 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.

8.8.18 ScaleFactorChanged(NewFactor as double)

Plugin Version: 22.1, Platform: macOS, Targets: .

Function: The backing store scale factor has changed.
Notes: Please invalidate any cached bitmaps or other relevant state.

8.8.19 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.

8.9 class IKImageViewMBS

8.9.1 class IKImageViewMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: An advanced image view in ImageKit.
Notes: The IKImageViewMBS class provides an efficient way to display images in a view while at the same time supporting a number of image editing operations such as rotating, zooming, and cropping. It supports drag and drop for the NSFilenamesPboardType flavor so that the user can drag an image to the view. If possible, image rendering uses hardware acceleration to achieve optimal performance. The IKImageViewMBS class is implemented as a subclass of NSView. Similar to NSImageViewMBS, the IKImageViewMBS class is used to display a single image.
Subclass of the NSViewMBS class.
Blog Entries

- News from the MBS Xojo Plugins Version 22.1

8.9.2 Methods

8.9.3 Constructor

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Creates a new box view with size 100/100 and position 0/0
Example:

```
dim x as new IKImageViewMBS
```

Notes: On success the handle property is not zero.
See also:

- 8.9.4 Constructor(Handle as Integer) 1109
- 8.9.5 Constructor(left as double, top as double, width as double, height as double) 1110

8.9.4 Constructor(Handle as Integer)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Creates an object based on the given NSView handle.
Example:

```
dim t as new IKImageViewMBS(0, 0, 100, 100)
dim v as new IKImageViewMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a IKImageView and the plugin retains this handle.
See also:

- 8.9.3 Constructor 1109
- 8.9.5 Constructor(left as double, top as double, width as double, height as double) 1110

8.9.5 Constructor(left as double, top as double, width as double, height as double)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

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 IKImageViewMBS(left, top, width, height)
```

Notes: On success the handle property is not zero.
See also:

- 8.9.3 Constructor 1109
- 8.9.4 Constructor(Handle as Integer) 1109

8.9.6 convertImagePointToViewPoint(imagePoint as NSPointMBS) as NSPointMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Converts an image coordinate to an image view coordinate.
Notes: imagePoint: A point specified in coordinates relative to the image.

Returns a point specified in coordinates relative to the image view.

8.9.7 convertImageRectToViewRect(ImageRect as NSRectMBS) as NSRectMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Converts an image rectangle to an image view rectangle.
Notes: ImageRect: An rectangle specified in coordinates relative to the image.

Returns an rectangle specified in coordinates relative to the image view.

8.9.8 convertViewPointToImagePoint(viewPoint as NSPointMBS) as NSPointMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Converts an image view coordinate to an image coordinate.
Notes: viewPoint: A point specified in coordinates relative to the image view.

Returns the point specified in coordinates relative to the image.

8.9.9 convertViewRectToImageRect(viewRect as NSRectMBS) as NSRectMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Converts an image view rectangle to an image rectangle.
Notes: viewRect: An rectangle specified in coordinates relative to the image view.

The rectangle specified in coordinates relative to the image.

8.9.10 crop

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Crops the image using the current selection.

8.9.11 flipImageHorizontal

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Flips an image along the horizontal axis.

8.9.12 flipImageVertical

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Flips an image along the vertical axis.

8.9.13 IKOverlayTypeBackground as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: A layer level.
Notes: A background.

8.9.14 IKOverlayTypeImage as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: A layer level.
Notes: An image.

8.9.15 IKToolModeAnnotate as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The annotation tool.

8.9.16 IKToolModeCrop as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The crop tool.

8.9.17 IKToolModeMove as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The move tool.

8.9.18 IKToolModeNone as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: No tool is set.

8.9.19 IKToolModeRotate as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The rotation tool.

8.9.20 IKToolModeSelect as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The selection tool.

8.9.21 IKToolModeSelectEllipse as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The selection ellipse.

8.9.22 IKToolModeSelectLasso as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: The selection lasso.

8.9.23 IKToolModeSelectRect as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: One of the tool modes.
Notes: Same as IKToolModeSelect.

8.9.24 rotateImageLeft

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Rotates the image left (counter-clockwise).

8.9.25 rotateImageRight

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Rotates the image right (clockwise).

8.9.26 scrollToPoint(point as NSPointMBS)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Scrolls the view to the specified point.
Notes: point: The point to scroll to.

8.9.27 scrollToRect(rect as NSRectMBS)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Scrolls the view so that it includes the provided rectangular area.
Notes: Rect: The rectangular area to include in the view.

8.9.28 SetImage(image as CGImageMBS, metaData as Dictionary = nil)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Sets the image to display in an image view.
Notes: image: The image to set.
metaData: A dictionary that contains metadata that describes the image.

8.9.29 SetImageFile(File as FolderItem)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Initializes an image view with the image specified by a folderitem.
Notes: File: The file that specifies the location of the image.

This method is the preferred initializer for RAW images. If you use this method for a TIFF file that contains multiple images, only the first image is displayed.

8.9.30 SetImageURL(URL as String)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Initializes an image view with the image specified by a URL.
Notes: url: The URL that specifies the location of the image.

This method is the preferred initializer for RAW images. If you use this method for a TIFF file that contains multiple images, only the first image is displayed.

8.9.31 setImageZoomFactor(zoomFactor as Double, centerPoint as NSPointMBS)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Sets the zoom factor at the provided origin.
Notes: zoomFactor: The zoom factor to apply to the image.
centerPoint: The point that specifies the origin of the zoom factor.

8.9.32 SetPicture(picture as Picture, metaData as Dictionary = nil)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Sets the picture to display in an image view.
Notes: picture: The picture to set.
metaData: A dictionary that contains metadata that describes the image.

8.9.33 setRotationAngle(rotationAngle as Double, centerPoint as NSPointMBS)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Sets the rotation angle at the provided origin.
Notes: rotationAngle: The rotation angle to apply to the image.
centerPoint: The point that specifies the origin of the rotation angle.

8.9.34 zoomImageToActualSize

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Zooms the image so that it is displayed using its true size.

8.9.35 zoomImageToFit

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Zooms the image so that it fits in the image view.

8.9.36 zoomImageToRect(rect as NSRectMBS)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Zooms the image so that it fits in the specified rectangle.
Notes: rect: The rectangle to fit the image in.

8.9.37 zoomIn

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Zooms the image in.

8.9.38 zoomOut

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Zooms the image out.

8.9.39 Properties

8.9.40 autohidesScrollers as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the automatic-hiding scroll bar state for the image view.
Notes: (Read and Write property)

8.9.41 autoresizes as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the automatic resizing state for the image view.
Notes: (Read and Write property)

8.9.42 backgroundColor as NSColorMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the background color for the image view.
Notes: (Read and Write property)

8.9.43 currentToolMode as String

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the current tool mode for the image view.
Notes: (Read and Write property)

8.9.44 doubleClickOpensImageEditPanel as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the image-opening state of the editing pane in the image view.
Notes: (Read and Write property)

8.9.45 editable as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the editable state for the image view.
Notes: (Read and Write property)

8.9.46 hasHorizontalScroller as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the horizontal scroll bar state for the image view.
Notes: (Read and Write property)

8.9.47 hasVerticalScroller as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the vertical scroll bar state for the image view.
Notes: (Read and Write property)

8.9.48 Image as CGImageMBS

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Returns the image associated with the view, after any image corrections.
Notes: (Read only property)

8.9.49 imageCorrection as Variant

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies a Core Image filter for image correction.
Notes: (Read and Write property)

8.9.50 imageHeight as Integer

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Returns the height of the image in the image view.
Notes: The image size changes whenever an image is rotates or cropped.
(Read only property)

8.9.51 imageProperties as Dictionary

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Returns the metadata for the image in the view.
Notes: A dictionary of metadata that specifies the image properties.
(Read only property)

8.9.52 imageWidth as Integer

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Returns the width of the image in the image view.
Notes: The image size changes whenever an image is rotates or cropped.
(Read only property)

8.9.53 Picture as Picture

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Returns the image associated with the view, after any image corrections.
Notes: Converted to picture for your convenience.
(Read only property)

8.9.54 rotationAngle as Double

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the rotation angle for the image view.
Notes: (Read and Write property)

8.9.55 supportsDragAndDrop as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the drag-and-drop support state for the image view.
Notes: (Read and Write property)

8.9.56 zoomFactor as Double

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the zoom factor for the image view.
Notes: (Read and Write property)

8.9.57 Overlay(layerType as string) as Variant

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Get or set the overlay layer.
Notes: The Core Animation layer associated with a layer type.
Value is a CALayerMBS object.
(Read and Write computed property)

8.10 class IKPictureTakerMBS

8.10.1 class IKPictureTakerMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: An IKPictureTaker object is a panel that allows users to choose and crop an image.
Notes: It supports browsing of the file system and includes a recents popup-menu. The IKPictureTaker lets the user to crop a choosen image or to take snapshot from a camera like the built-in iSight.

Requires Mac OS X 10.5.
Subclass of the NSPanelMBS class.

8.10.2 Methods

8.10.3 Available as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether the picture taker is available on that platform or not.
Example:

```
dim n as Integer

dim p as new IKPictureTakerMBS

if not p.Available then
MsgBox "This application requires Mac OS X 10.5 and a Macho Target"
Return
end if
```

Notes: True on Mac OS X 10.5.

8.10.4 beginPictureTaker as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Launch the PictureTaker.
Example:

```
dim p as IKPictureTakerMBS // your picture taker
```

```
if not p.beginPictureTaker then
MsgBox "Can't show picture taker!?"
end if
```

Notes: You will later receive an event for the case the user clicks on OK or Cancel buttons.
Returns true on success and false on failure.

8.10.5 beginPictureTakerSheet(parent as DesktopWindow) as boolean

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

Function: Opens a picture taker as a sheet whose parent is the specified window.
Notes: parent: The parent window of the picture taker sheet.

You will later receive an event for the case the user clicks on OK or Cancel buttons.

Available in Mac OS X v10.5 and later.
See also:

- 8.10.6 beginPictureTakerSheet(parent as NSWindowMBS) as boolean 1122
- 8.10.7 beginPictureTakerSheet(parent as window) as boolean 1122

8.10.6 beginPictureTakerSheet(parent as NSWindowMBS) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Opens a picture taker as a sheet whose parent is the specified window.
Notes: parent: The parent window of the picture taker sheet.

You will later receive an event for the case the user clicks on OK or Cancel buttons.

Available in Mac OS X v10.5 and later.
See also:

- 8.10.5 beginPictureTakerSheet(parent as DesktopWindow) as boolean 1122
- 8.10.7 beginPictureTakerSheet(parent as window) as boolean 1122

8.10.7 beginPictureTakerSheet(parent as window) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Opens a picture taker as a sheet whose parent is the specified window.
Notes: parent: The parent window of the picture taker sheet.

You will later receive an event for the case the user clicks on OK or Cancel buttons.

Available in Mac OS X v10.5 and later.
See also:

- 8.10.5 beginPictureTakerSheet(parent as DesktopWindow) as boolean 1122
- 8.10.6 beginPictureTakerSheet(parent as NSWindowMBS) as boolean 1122

8.10.8 Constructor

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new picture taker panel.

8.10.9 CropAreaSizeHeight as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The height of the crop area.

8.10.10 CropAreaSizeWidth as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The width of the crop area.

8.10.11 outputImage as NSImageMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Return the edited image.

8.10.12 OutputImageMaxSizeKeyHeight as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The maximum height of the output image.

8.10.13 OutputImageMaxSizeKeyWidth as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The maximum width of the output image.

8.10.14 popUpRecentsMenuForView(parent as NSViewMBS) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Displays the Open Recent popup menu associated with the picture taker.
Notes: You will later receive an event for the case the user clicks on OK or Cancel buttons.

Available in Mac OS X v10.5 and later.

8.10.15 runModal as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Launches a modal PictureTaker session.
Example:

```
dim p as IKPictureTakerMBS // global property
dim n as Integer

p=new IKPictureTakerMBS

if not p.Available then
MsgBox "This application requires Mac OS X 10.5 and a Macho Target"
Return
end if

p.AllowsFileChoosing=true
p.AllowsEditing=true
p.AllowsVideoCapture=true
p.ShowEffects=FALSE // disable if you run modal!
p.ShowRecentPicture=true
p.UpdateRecentPicture=true
p.InformationalText="Please take a picture"

n=p.runModal
```

```
if n=1 then // ok
Backdrop=p.outputImage.CopyPictureWithMask
else
Title=Str(n)
end if
```

Notes: Returns NSOKButton (1) if the user edits or chooses an image and confirm panel, NSCancelButton (0) if the user canceled or didn't change the image.

You may want to disable effects as they won't work in Xojo in a modal picture taker dialog.

8.10.16 SetCropAreaSize(width as Double, height as Double)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Sets the crop area.

8.10.17 SetOutputImageMaxSize(width as Double, height as Double)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Sets the maximum output image size.

8.10.18 Properties

8.10.19 AllowsEditing as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether editing is allowed or not.
Notes: (Read and Write computed property)

8.10.20 AllowsFileChoosing as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether choosing a file is allowed or not.
Notes: (Read and Write computed property)

8.10.21 AllowsVideoCapture as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether video capture is allowed or not.
Notes: (Read and Write computed property)

8.10.22 InformationalText as NSAttributedStringMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The information text as a plain string.
Notes: On getting the value the plugin will check whether the value is a formatted or a plain text. if it is a plain text, it will return the plain text as a NSAttributedStringMBS.
(Read and Write computed property)
See also:

- 8.10.23 InformationalText as string 1126

8.10.23 InformationalText as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The information text as a plain string.
Example:

```
dim p as IKPictureTakerMBS // your picture taker
p.InformationalText="Please take a picture"
```

Notes: On getting the value the plugin will check whether the value is a formatted or a plain text. if it is a formatted text, it will return the formatted text as plain text.
(Read and Write computed property)
See also:

- 8.10.22 InformationalText as NSAttributedStringMBS 1126

8.10.24 inputImage as NSImageMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The input image.
Notes: The input image is never modified by the PictureTaker.
(Read and Write computed property)

8.10.25 mirroring as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if video mirroring is enabled, false otherwise.
Notes: Controls whether the receiver enable/disable video mirroring durring snapshots (default is true).
(Read and Write computed property)

8.10.26 RemainOpenAfterValidate as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the picture taker reamins open.
Notes: Requires Mac OS X 10.6.
(Read and Write computed property)

8.10.27 ShowAddressBookPicture as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether the addressbook picture is shown or not.
Notes: (Read and Write computed property)

8.10.28 ShowEffects as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether effects are shown or not.
Notes: (Read and Write computed property)

8.10.29 ShowEmptyPicture as NSImageMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The image to use for an empty image.
Notes: (Read and Write computed property)

8.10.30 ShowRecentPicture as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether recent pictures should be shown.
Notes: (Read and Write computed property)

8.10.31 UpdateRecentPicture as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether recent pictures should be updated.
Notes: (Read and Write computed property)

8.10.32 Events

8.10.33 Finished(returnCode as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: beginPictureTaker has finished work.
Notes: ReturnCode is 1 if the user clicked OK and 0 if the user clicked false.

8.11 class IKSlideshowMBS

8.11.1 class IKSlideshowMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The ImageKit class for a slideshow.
Notes: Requires Mac OS X 10.5.
Slideshows can be only with pictures, with PDF pages or with file references.
Those files can be picture files, pdf files or anything you want.
Blog Entries

- MBS REALbasic plug-in 9.6

8.11.2 Methods

8.11.3 addFile(file as folderitem, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Adds a file to the items list.

8.11.4 addImage(image as NSImageMBS, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Adds an image to the items list.
Example:

```
dim p as picture
dim n as NSImageMBS
dim s as new IKSlideshowMBS

// get picture to p
n=new NSImageMBS(p)
s.addImage n
```

8.11.5 addPage(page as Variant, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Adds a PDF page to the items list.
Notes: Page must be a PDFPageMBS object.

8.11.6 Available as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether the slide show functions are available.
Example:

```
if IKSlideshowMBS.Available=False then
MsgBox "You need Mac OS X 10.5 for this and a MachO application."
quit
end if
```

Notes: Value is true for Mac OS X 10.5.

8.11.7 canExportToApplication(applicationBundleIdentifier as string) as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: is exporting to a given application possible?
Notes: (application installed?, right version?, ...)

8.11.8 exportSlideshowItems(applicationBundleIdentifier as string)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Export items to the given application.
Example:

```
dim i,c as Integer
dim s as new IKSlideshowMBS

// add items here

if false=IKSlideshowMBS.canExportToApplication(IKSlideshowMBS.iPhotoBundleIdentifier) then
MsgBox "Can't export to iPhoto."
else

if s.itemcount>0 then

s.exportSlideshowItems IKSlideshowMBS.iPhotoBundleIdentifier
```

```
else
MsgBox "no slides?"
end if
end if
```

8.11.9 indexOfCurrentSlideshowItem as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The index of the current slide.
Notes: Index is from 0 to count-1.

8.11.10 ItemCount as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the number of items.

8.11.11 reloadData

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Reloads all slides.

8.11.12 reloadSlideshowItemAtIndex(index as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Reloads the slide show with the given index.

8.11.13 removeItem(index as Integer)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Removes an item from the item list.

8.11.14 removeItems

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Removes all items from the item list.

8.11.15 runSlideshow

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Start the slideshow.
Example:

```
dim s as new IKSlideshowMBS
// add items
s.runSlideshow
```

Notes: You may want to set all the properties before.

8.11.16 setFile(index as Integer, file as folderitem, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the file in the items list with the given index.

8.11.17 setImage(index as Integer, image as NSImageMBS, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets the image in the items list with the given index.

8.11.18 setPage(index as Integer, page as Variant, name as string="")

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets a PDF page in the items list with the given index.
Notes: Page must be a PDFPageMBS object.

8.11.19 stopSlideshow

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Stops the slideshow.

8.11.20 Properties

8.11.21 AudioFile as FolderItem

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Optional audio file to play while running slide show.
Notes: Only used for Mac OS X 10.6.
(Read and Write property)

8.11.22 PDFDisplayBox as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The PDF display box mode to use.
Notes: Default value is -1 which means that we use the framework default mode.
(Read and Write property)

8.11.23 PDFDisplayMode as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The PDF display mode you want.
Notes: Default value is -1 which means that we use the framework default mode.
(Read and Write property)

8.11.24 PDFDisplaysAsBook as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether PDF should display as book.
Notes: Default value is false.
(Read and Write property)

8.11.25 ScreenIndex as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The screen to use.
Notes: Default is main screen.
Only used for Mac OS X 10.6.
(Read and Write property)

8.11.26 StartIndex as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The index of the first slide to show.
Notes: Index is from 0 to count-1.
Default value is -1 which means that we use the framework default mode.
(Read and Write property)

8.11.27 StartPaused as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether to start paused.
Notes: Default is false.
(Read and Write property)

8.11.28 WrapAround as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether to wrap around when the slideshow runs.
Notes: Default is false.
(Read and Write property)

8.11.29 autoPlayDelay as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The time to wait before the slideshow will start automatically.
Notes: Value is in seconds.
(Read and Write computed property)

8.11.30 Events

8.11.31 canExportSlideshowItemAtIndex(index as Integer, applicationBundleIdentifier as string) as boolean

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: Should the export button be enabled for a given item at index?
Notes: This event is optional.

8.11.32 slideshowDidChangeCurrentIndex(newIndex as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: Slideshow did change current item index.

8.11.33 slideshowDidStop

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: Slideshow did stop
Notes: This event is optional.

8.11.34 slideshowWillStart

Plugin Version: 7.7, Platform: macOS, Targets: .

Function: Slideshow will start.
Notes: This event is optional.

8.11.35 Constants

Constants

Constant	Value	Description
iPhotoBundleIdentifier	”com.apple.iPhoto”	The iPhoto application identifier. May be used on the export functions.
kPDFDisplayBoxArtBox	4	One of the possible values for the PDFDisplayBox property.
kPDFDisplayBoxBleedBox	2	One of the possible values for the PDFDisplayBox property.
kPDFDisplayBoxCropBox	1	One of the possible values for the PDFDisplayBox property.
kPDFDisplayBoxMediaBox	0	One of the possible values for the PDFDisplayBox property.
kPDFDisplayBoxTrimBox	3	One of the possible values for the PDFDisplayBox property.
kPDFDisplaySinglePage	0	One of the PDF display mode constants.
kPDFDisplaySinglePageContinuous	1	One of the PDF display mode constants.
kPDFDisplayTwoUp	2	One of the PDF display mode constants.
kPDFDisplayTwoUpContinuous	3	One of the PDF display mode constants.

Chapter 9

Midi

9.1 class AVMIDIPlayerMBS

9.1.1 class AVMIDIPlayerMBS

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: The AVMIDIPlayer class is a player for music file formats such as MIDI and iMelody.
Notes: Requires Mac OS X 10.10.

See also MidiPlaybackMBS class for playing notes.
Blog Entries

- MBS Xojo / Real Studio plug-ins version 16.2
- MBS Xojo / Real Studio Plugins, version 16.2pr1

9.1.2 Methods

9.1.3 Constructor(Data as MemoryBlock, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS)

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Initializes a newly allocated MIDI player with the contents of the String, using the specified sound bank.
Notes: Data: The data to play.
SoundBankFile: The folderitem of the sound bank. The sound bank must be a SoundFont2 or DLS bank.
Error: Returns, by-reference, a description of the error, if an error occurs.

For OS X the bank folderitem can be set to nil to use the default sound bank. However, iOS must always refer to a valid bank file.
See also:

- 9.1.4 Constructor(Data as String, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1138
- 9.1.5 Constructor(File as FolderItem, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1138

9.1.4 Constructor(Data as String, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS)

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Initializes a newly allocated MIDI player with the contents of the String, using the specified sound bank.
Notes: Data: The data to play.
SoundBankFile: The folderitem of the sound bank. The sound bank must be a SoundFont2 or DLS bank.
Error: Returns, by-reference, a description of the error, if an error occurs.

For OS X the bank folderitem can be set to nil to use the default sound bank. However, iOS must always refer to a valid bank file.
See also:

- 9.1.3 Constructor(Data as MemoryBlock, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1137
- 9.1.5 Constructor(File as FolderItem, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1138

9.1.5 Constructor(File as FolderItem, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS)

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Initializes a newly allocated MIDI player with the contents of the file, using the specified sound bank.
Notes: File: The file to play.
SoundBankFile: The folderitem of the sound bank. The sound bank must be a SoundFont2 or DLS bank.
Error: Returns, by-reference, a description of the error, if an error occurs.

For OS X the bank folderitem can be set to nil to use the default sound bank. However, iOS must always refer to a valid bank file.
See also:

- 9.1.3 Constructor(Data as MemoryBlock, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1137
- 9.1.4 Constructor(Data as String, SoundBankFile as FolderItem = nil, byref error as NSErrorMBS) 1138

9.1.6 Destructor

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: The destructor.

9.1.7 play

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Plays the sequence.
Notes: If prepareToPlay has not been invoked, play may be delayed while the events are prerolled.

9.1.8 prepareToPlay

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Prepares to play the sequence by prerolling all events.
Notes: This happens automatically on play if it has not already been called, but may produce a delay in startup.

9.1.9 stop

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Stops playing the sequence.

9.1.10 Properties

9.1.11 CurrentPosition as Double

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: The current playback position, in seconds.
Notes: You can set the currentPosition of the player while the player is playing, in which case playback will

resume at the new time.

Note: No range checking on the currentPosition value is done when you set it to a new value. It is your responsibility to ensure the position is valid relative to the duration.
(Read and Write property)

9.1.12 Duration as Double

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: The length of the currently loaded file, in seconds.
Notes: (Read only property)

9.1.13 Handle as Integer

Plugin Version: 16.2, Platform: macOS, Targets: All.

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

9.1.14 Playing as Boolean

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: Whether the sequence is playing.
Notes: The player may have reached the end of all the events in any of its tracks, but it will return true until it is stopped.
(Read only property)

9.1.15 Rate as Double

Plugin Version: 16.2, Platform: macOS, Targets: All.

Function: The playback rate of the player.
Notes: The default value is 1.0, normal playback rate.
(Read and Write property)

9.1.16 Events

9.1.17 Completed

Plugin Version: 16.2, Platform: macOS, Targets: .

Function: Called when a MIDI playback request is completed.

Chapter 10

Speech

10.1 class AVSpeechSynthesisMarkerMBS

10.1.1 class AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: An object that contains information about the synthesized audio.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.1.2 Methods

10.1.3 Constructor

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The private constructor.

10.1.4 copy as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a copy of the object.

10.1.5 markerWithBookmarkName(mark as string, atByteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a bookmark marker with a name and offset into the audio buffer.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

m = AVSpeechSynthesisMarkerMBS.markerWithBookmarkName("test", 1234)

MessageBox m.bookmarkName+" at "+m.byteSampleOffset.ToString
```

Notes: mark: The name of the bookmark.
byteSampleOffset: The byte offset into the audio buffer.

10.1.6 markerWithMarkerType(type as integer, TextRange as NSRangeMBS, byteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a marker with a type and location of the request‚Äôs text.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

Dim range As New NSRangeMBS(5, 6)
m = AVSpeechSynthesisMarkerMBS.markerWithMarkerType(AVSpeechSynthesisMarkerMBS.SpeechSynthesisMarkerMarkWord, range, 1234)

MessageBox m.textRange.String+" at "+m.byteSampleOffset.ToString
```

Notes: type: The type that describes the text.
range: The location and length of the request‚Äôs text.
byteSampleOffset: The byte offset into the audio buffer.

10.1.7 markerWithParagraphRange(TextRange as NSRangeMBS, byteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a paragraph marker with a range of the paragraph and offset into the audio buffer.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

Dim range As New NSRangeMBS(5, 6)
m = AVSpeechSynthesisMarkerMBS.markerWithParagraphRange(range, 1234)

MessageBox m.textRange.String+” at ”+m.byteSampleOffset.ToString
```

Notes: range: The location and length of the paragraph.
byteSampleOffset: The byte offset into the audio buffer.

10.1.8 markerWithPhonemeString(phoneme as string, atByteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a phoneme marker with a range of the phoneme and offset into the audio buffer.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

m = AVSpeechSynthesisMarkerMBS.markerWithPhonemeString(”I”, 1234)

MessageBox m.phoneme+” at ”+m.byteSampleOffset.ToString
```

Notes: phoneme: A string that represents a distinct sound.
byteSampleOffset: The byte offset into the audio buffer.

10.1.9 markerWithSentenceRange(TextRange as NSRangeMBS, byteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a sentence marker with a range of the sentence and offset into the audio buffer.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

Dim range As New NSRangeMBS(5, 6)
m = AVSpeechSynthesisMarkerMBS.markerWithSentenceRange(range, 1234)

MessageBox m.textRange.String+” at ”+m.byteSampleOffset.ToString
```

Notes: range: The location and length of the word.

byteSampleOffset: The byte offset into the audio buffer.

10.1.10 markerWithWordRange(TextRange as NSRangeMBS, byteSampleOffset as Integer) as AVSpeechSynthesisMarkerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a word marker with a range of the word and offset into the audio buffer.
Example:

```
Dim m As AVSpeechSynthesisMarkerMBS

Dim range As New NSRangeMBS(5, 6)
m = AVSpeechSynthesisMarkerMBS.markerWithWordRange(range, 1234)

MessageBox m.textRange.String+" at "+m.byteSampleOffset.ToString
```

Notes: range: The location and length of the word.
byteSampleOffset: The byte offset into the audio buffer.

10.1.11 Properties

10.1.12 bookmarkName as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A string that represents the name of a bookmark.
Notes: (Read and Write property)

10.1.13 byteSampleOffset as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The byte offset into the audio buffer.
Notes: (Read and Write property)

10.1.14 Handle as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

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

10.1.15 mark as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The type that describes the text.
Notes: (Read and Write property)

10.1.16 phoneme as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A string that represents a distinct sound.
Notes: (Read and Write property)

10.1.17 textRange as NSRangeMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The location and length of the request,Äôs text.
Notes: (Read and Write property)

10.1.18 Constants

Marker Types

Constant	Value	Description
SpeechSynthesisMarkerMarkBookmark	4	A type of text that represents a bookmark.
SpeechSynthesisMarkerMarkParagraph	3	A type of text that represents a paragraph.
SpeechSynthesisMarkerMarkPhoneme	0	A type of text that represents a phoneme.
SpeechSynthesisMarkerMarkSentence	2	A type of text that represents a sentence.
SpeechSynthesisMarkerMarkWord	1	A type of text that represents a word.

10.2 class AVSpeechSynthesisVoiceMBS

10.2.1 class AVSpeechSynthesisVoiceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A distinct voice for use in speech synthesis.
Notes: The primary factors that distinguish a voice in speech synthesis are language, locale, and quality. Create an instance of AVSpeechSynthesisVoice to select a voice that‚Äôs appropriate for the text and the language, and set it as the value of the voice property on an AVSpeechUtterance instance. The voice may optionally reflect a local variant of the language, such as Australian or South African English.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.2.2 Methods

10.2.3 Constructor

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The private constructor.

10.2.4 SpeechUtteranceDefaultSpeechRate as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The default speech rate.

10.2.5 SpeechUtteranceMaximumSpeechRate as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The maximum speech rate.

10.2.6 SpeechUtteranceMinimumSpeechRate as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The minimum speech rate.

10.2.7 speechVoices as AVSpeechSynthesisVoiceMBS()

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Retrieves all available voices on the device.
Example:

```
Dim voices() As AVSpeechSynthesisVoiceMBS = AVSpeechSynthesisVoiceMBS.speechVoices

Dim Names() As String
For Each voice As AVSpeechSynthesisVoiceMBS In voices
names.append voice.name
Next

MessageBox Join(names, ", ")
```

Notes: Use the language property to identify each voice by its language and locale.

10.2.8 VoiceIdentifierAlex as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The voice that the system identifies as Alex.
Notes: The Alex voice is only available for the en-US language code.

10.2.9 voiceWithIdentifier(identifier as string) as AVSpeechSynthesisVoiceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Retrieves a voice for the identifier you specify.
Example:

```
Dim voice As AVSpeechSynthesisVoiceMBS = AVSpeechSynthesisVoiceMBS.voiceWithIdentifier("com.apple.voice.compact.en-GB.Daniel")

MessageBox voice.name
```

Notes: identifier: The unique identifier for a voice.

A voice for the specified identifier if the identifier is valid and the voice is available on the device; otherwise, nil.

10.2.10 voiceWithLanguage(languageCode as string) as AVSpeechSynthesisVoiceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Retrieves a voice for the BCP 47 code language code you specify.
Example:

```
Dim voice As AVSpeechSynthesisVoiceMBS = AVSpeechSynthesisVoiceMBS.voiceWithLanguage(”en-GB”)

MessageBox voice.name
```

Notes: language: A BCP 47 code that identifies the language and locale for a voice.

Returns a voice for the specified language and locale code if the code is valid; otherwise, nil.

Pass ”” for languageCode to receive the default voice for the system,Äôs language and region.

10.2.11 Properties

10.2.12 audioFileSettings as Dictionary

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A dictionary that contains audio file settings.
Notes: If you want to generate speech and save it as an audio file to share or play later, use this dictionary to create an AVAudioFile instance and pass it as the settings parameter.
You can determine the AVAudioCommonFormat and interleaved properties of a voice from this dictionary.
The format of this dictionary matches the data that SpeechSynthesizerBuffer event provides for the same voice.
(Read only property)

10.2.13 currentLanguageCode as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the language and locale code for the user,Äôs current locale.
Example:

```
MessageBox AVSpeechSynthesisVoiceMBS.currentLanguageCode
```

Notes: A string that contains the BCP 47 language and locale code for the user,Äôs current locale.
(Read only property)

10.2.14 gender as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The gender for a voice.
Notes: (Read only property)

10.2.15 Handle as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

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

10.2.16 identifier as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The unique identifier of a voice.
Notes: (Read only property)

10.2.17 language as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A BCP 47 code that contains the voice‚Äôs language and locale.
Notes: The language of a voice controls the conversion of text to spoken phonemes. For best results, ensure that the language of an utterance‚Äôs text matches the voice for the utterance. The locale of a voice reflects regional variations in pronunciation or accent. For example, a voice with a language code of en-US speaks English text with a North American accent, and a language code of en-AU speaks English text with an Australian accent.
(Read only property)

10.2.18 name as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The name of a voice.
Notes: (Read only property)

10.2.19 quality as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The speech quality of a voice.
Notes: (Read only property)

10.2.20 voiceTraits as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The traits of a voice.
Notes: (Read only property)

10.2.21 Constants

Gender

Constant	Value	Description
SpeechSynthesisVoiceGenderFemale	2	The female voice option.
SpeechSynthesisVoiceGenderMale	1	The male voice option.
SpeechSynthesisVoiceGenderUnspecified	0	The nonspecific gender option.

Quality

Constant	Value	Description
SpeechSynthesisVoiceQualityDefault	1	A basic quality voice that‚Äôs available on the device by default.
SpeechSynthesisVoiceQualityEnhanced	2	An enhanced quality voice that you must download to use.
SpeechSynthesisVoiceQualityPremium	3	A premium quality voice that you must download to use.

Voice Traits

Constant	Value	Description
SpeechSynthesisVoiceTraitIsNoveltyVoice	1	The trait that indicates a voice is a novelty voice. An example of a novelty voice is a character‚Äôs voice in a game.
SpeechSynthesisVoiceTraitIsPersonalVoice	2	The trait that indicates a voice is a personal voice. A user generates and owns a personal voice.
SpeechSynthesisVoiceTraitNone	0	The trait that indicates a voice is a regular voice.

10.3 class AVSpeechSynthesizerMBS

10.3.1 class AVSpeechSynthesizerMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: An object that produces synthesized speech from text utterances and enables monitoring or controlling of ongoing speech.
Notes: To speak some text, create an AVSpeechUtteranceMBS instance that contains the text and pass it to speakUtterance on a speech synthesizer instance. You can optionally also retrieve an AVSpeechSynthesisVoiceMBS and set it on the utterance,Äôs voice property to have the speech synthesizer use that voice when speaking the utterance,Äôs text.

The speech synthesizer maintains a queue of utterances that it speaks. If the synthesizer isn,Äôt speaking, calling speakUtterance: begins speaking that utterance either immediately or after pausing for its preUtteranceDelay, if necessary. If the synthesizer is speaking, the synthesizer adds utterances to a queue and speaks them in the order it receives them.
After speech begins, you can use the synthesizer object to pause or stop speech. After pausing, you can resume the speech from its paused point or stop the speech entirely and remove all remaining utterances in the queue.
You can monitor the speech synthesizer by examining its speaking and paused properties, or by adding events.

An AVSpeechSynthesizerMBS also controls the route where the speech plays. For more information, see Directing speech output.

The system doesn,Äôt automatically retain the speech synthesizer, so you need to manually retain it until speech concludes.
Blog Entries

- MBS Xojo Plugins, version 24.1pr2

10.3.2 Methods

10.3.3 Constructor

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The constructor.

10.3.4 continueSpeaking as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Resumes speech from its paused point.
Notes: Returns true if speech resumes; otherwise, false.

This method only has an effect if the speech synthesizer is in a paused state.

10.3.5 Destructor

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The destructor.

10.3.6 IPANotationAttribute as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A string that contains International Phonetic Alphabet (IPA) symbols the speech synthesizer uses to control pronunciation of certain words or phrases.
Notes: For example, the speech synthesizer uses an AVSpeechSynthesisIPANotationAttribute instance to control pronunciation of a proper name.

10.3.7 pauseSpeakingAtBoundary(SpeechBoundary as Integer) as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Pauses speech at the boundary you specify.
Notes: boundary: An enumeration that describes whether to pause speech immediately or only after the synthesizer finishes speaking the current word.

Returns true if speech pauses; otherwise, false.

The boundary parameter also affects how the speech synthesizer resumes speaking text after a pause and call to continueSpeaking. If the boundary is SpeechBoundaryImmediate, speech resumes from the exact point where it pauses, even if that point occurs in the middle of speaking a word. If the boundary is SpeechBoundaryWord, speech resumes from the word that follows the last spoken word where it pauses.

10.3.8 requestPersonalVoiceAuthorization

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Prompts the user to authorize your app to use personal voices.

10.3.9 speakUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Adds the utterance you specify to the speech synthesizer,Äôs queue.
Notes: utterance: An AVSpeechUtteranceMBS instance that contains text to speak.
Discussion

Warning: Attempting to enqueue the same utterance more than once throws an exception.

10.3.10 stopSpeakingAtBoundary(SpeechBoundary as Integer) as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Stops speech at the boundary you specify.
Notes: boundary: An enumeration that describes whether to stop speech immediately or only after the synthesizer finishes speaking the current word.

Returns true if speech stops; otherwise, false.

Unlike pausing a speech synthesizer, which can resume after a pause, stopping the synthesizer immediately cancels speech and removes all unspoken utterances from the synthesizer,Äôs queue.

10.3.11 writeUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Generates speech for the utterance and invokes the callback with the audio buffer.
Notes: utterance: The utterance for synthesizing speech.

Call this method to receive audio buffers to store or further process synthesized speech.

Calls SpeechSynthesizerBuffer and SpeechSynthesizerMarker events later.

10.3.12 Properties

10.3.13 Handle as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

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

10.3.14 isPaused as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A Boolean value that indicates whether a speech synthesizer is in a paused state.
Notes: If true, the speech synthesizer is in a paused state after beginning to speak an utterance; otherwise, false.
(Read only property)

10.3.15 isSpeaking as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A Boolean value that indicates whether the speech synthesizer is speaking or is in a paused state and has utterances to speak.
Notes: If true, the synthesizer is speaking or is in a paused state with utterances in its queue. If false, the synthesizer isn‚Äôt speaking and it doesn‚Äôt have any utterances in its queue.
(Read only property)

10.3.16 mixToTelephonyUplink as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: iOS only.

Function: A Boolean value that specifies whether to send synthesized speech to an active call.
Notes: This property has no effect when there isn‚Äôt an active call.
(Read and Write property)

10.3.17 personalVoiceAuthorizationStatus as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Your app‚Äôs authorization to use personal voices.

Notes: The user can grant or deny your app‚Äôs request to use personal voices when they‚Äôre initially prompted, and change the authorization in the Settings app. Additionally, the framework denies the request if the device doesn‚Äôt support using personal voices.
(Read only property)

10.3.18 usesApplicationAudioSession as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: iOS only.

Function: A Boolean value that specifies whether the app manages the audio session.
Notes: If you set this value to false, the system creates a separate audio session to automatically manage speech, interruptions, and mixing and ducking the speech with other audio sources.
(Read and Write property)

10.3.19 Events

10.3.20 availableVoicesDidChange

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: An event that indicates a change in available voices for speech synthesis.
Notes: The system posts this notification when available voices for speech synthesis on the system change. For example, a new personal voice becomes available and the user authorized the app to access personal voices. Or new 3rd party voices become available through an app the user downloads.

10.3.21 didCancelSpeechUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer cancels speaking an utterance.
Notes: utterance: The utterance that the speech synthesizer cancels speaking.

The system only calls this method if a speech synthesizer is speaking an utterance and the system calls its stopSpeakingAtBoundary method. The system doesn‚Äôt call this method if the synthesizer is in a delay between utterances when speech stops, and it doesn‚Äôt call it for unspoken utterances.

10.3.22 didContinueSpeechUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer resumes speaking an utterance after pausing.
Notes: utterance: The utterance that the speech synthesizer resumes speaking.

The system only calls this method if a speech synthesizer pauses speaking and the system calls its pauseSpeakingAtBoundary method. The system doesn‚Äôt call this method if the synthesizer pauses while in a delay between utterances.

10.3.23 didFinishSpeechUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer finishes speaking an utterance.
Notes: utterance: The utterance that the speech synthesizer finishes speaking.

The system ignores the final utterance‚Äôs postUtteranceDelay and calls this method immediately when speech ends.

10.3.24 didPauseSpeechUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer pauses while speaking an utterance.
Notes: utterance: The utterance that the speech synthesizer pauses speaking.

The system only calls this method if a speech synthesizer is speaking an utterance and the system calls its pauseSpeakingAtBoundary method. The system doesn‚Äôt call this method if the synthesizer is in a delay between utterances when speech pauses.

10.3.25 didStartSpeechUtterance(utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer begins speaking an utterance.
Notes: utterance: The utterance that the speech synthesizer starts speaking.

If the utterance‚Äôs preUtteranceDelay property is greater than zero, the system calls this method after the delay completes and speech begins.

10.3.26 requestPersonalVoiceAuthorizationCompleted(status as Integer)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: A completion event that the system calls after the user responds to a request to authorize use of personal voices, which receives the authorization status as an argument.

10.3.27 SpeechSynthesizerBuffer(buffer as AVAudioBufferMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: The event that receives a buffer of generated speech.

10.3.28 SpeechSynthesizerMarker(markers() as AVSpeechSynthesisMarkerMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: The event that receives speech markers.

10.3.29 willSpeakMarker(marker as AVSpeechSynthesisMarkerMBS, utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer is about to speak a marker of an utterance‚Äôs text.
Notes: marker: The synthesized audio that the speech synthesizer is about to speak.
utterance: The utterance that the speech synthesizer pauses speaking.

10.3.30 willSpeakRangeOfSpeechString(characterRange as NSRangeMBS, utterance as AVSpeechUtteranceMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: .

Function: Tells you when the synthesizer is about to speak a portion of an utterance‚Äôs text.
Notes: characterRange: The range of characters in the utterance‚Äôs speechString that correspond to the unit of speech the synthesizer is about to speak.
utterance: The utterance that the speech synthesizer is about to speak.

The system calls this method once for each unit of speech in the utterance‚Äôs text, which is generally a word.

Tip
Implement this method if you want to provide a user interface to visually highlight each word as the synthesizer speaks it.

10.3.31 Constants

Speech Boundaries

Constant	Value	Description
SpeechBoundaryImmediate	0	Indicates to pause or stop speech immediately.
SpeechBoundaryWord	1	Indicates to pause or stop speech after the synthesizer finishes speaking the current word.

Authorization Status

Constant	Value	Description
SpeechSynthesisPersonalVoiceAuthorizationStatusAuthorized	3	The user granted your app,Äôs request to us
SpeechSynthesisPersonalVoiceAuthorizationStatusDenied	1	The user denied your app,Äôs request to use
SpeechSynthesisPersonalVoiceAuthorizationStatusNotDetermined	0	The app hasn,Äôt requested authorization to
SpeechSynthesisPersonalVoiceAuthorizationStatusUnsupported	2	The device doesn,Äôt support personal voice

10.4 class AVSpeechUtteranceMBS

10.4.1 class AVSpeechUtteranceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: An object that encapsulates the text for speech synthesis and parameters that affect the speech.
Notes: An AVSpeechUtterance is the basic unit of speech synthesis.

To synthesize speech, create an AVSpeechUtterance instance with text you want a speech synthesizer to speak. Optionally, change the voice, pitchMultiplier, volume, rate, preUtteranceDelay, or postUtteranceDelay parameters for the utterance. Pass the utterance to an instance of AVSpeechSynthesizer to begin speech, or enqueue the utterance to speak later if the synthesizer is already speaking.

Split a body of text into multiple utterances if you want to apply different speech parameters. For example, you can emphasize a sentence by increasing the pitch and decreasing the rate of that utterance relative to others, or you can introduce pauses between sentences by putting each into an utterance with a leading or trailing delay.

Set and use the AVSpeechSynthesizerMBS events to receive notifications when the synthesizer starts or finishes speaking an utterance. Create an utterance for each meaningful unit in a body of text if you want to receive notifications as its speech progresses.

10.4.2 Methods

10.4.3 Constructor(s as NSAttributedStringMBS)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates an utterance with the attributed text string that you specify for the speech synthesizer to speak.
Notes: To speak the text, pass the utterance to an instance of AVSpeechSynthesizerMBS.
See also:

- 10.4.4 Constructor(s as string) 1161

10.4.4 Constructor(s as string)

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates an utterance with the text string that you specify for the speech synthesizer to speak.
Example:

```
Dim s As New AVSpeechUtteranceMBS(”test”)

MessageBox s.speechString
```

See also:

- 10.4.3 Constructor(s as NSAttributedStringMBS) 1161

10.4.5 copy as AVSpeechUtteranceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a copy of the object.

10.4.6 speechUtteranceWithAttributedString(s as NSAttributedStringMBS) as AVSpeechUtteranceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates an utterance with the attributed text string that you specify for the speech synthesizer to speak.

10.4.7 speechUtteranceWithSSMLRepresentation(s as string) as AVSpeechUtteranceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Returns a new speech utterance with an Speech Synthesis Markup Language (SSML) string.
Notes: string: A string to speak that contains valid SSML markup. The initializer returns nil if you pass an invalid SSML string.

Returns a new speech utterance, or nil if the SSML string is invalid.

If using SSML to request voices that fall under certain attributes, the system may split a single utterance into multiple parts and send each to an appropriate synthesizer.
If no voice matches the properties, the utterance uses the voice set in its voice property. If you don,Äôt specify a voice, the system uses its default voice.

Note
Speech utterance properties that affect the prosody of a voice, such as its rate and pitchMultiplier, don,Äôt

apply to an utterance that uses an SSML representation.

10.4.8 speechUtteranceWithString(s as string) as AVSpeechUtteranceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: Creates an utterance with the text string that you specify for the speech synthesizer to speak.

10.4.9 Properties

10.4.10 attributedSpeechString as NSAttributedStringMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: An attributed string that contains the text for speech synthesis.
Notes: You can‚Äôt change an utterance‚Äôs text after initializaiton. If you want the speech synthesizer to speak different text, create a new utterance.
(Read only property)

10.4.11 Handle as Integer

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

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

10.4.12 pitchMultiplier as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The baseline pitch the speech synthesizer uses when speaking the utterance.
Notes: Before enqueing the utterance, set this property to a value within the range of 0.5 for lower pitch to 2.0 for higher pitch. The default value is 1.0. Setting this after enqueing the utterance has no effect.
(Read and Write property)

10.4.13 postUtteranceDelay as Double

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The amount of time the speech synthesizer pauses after speaking an utterance before handling the next utterance in the queue.
Notes: When multiple utterances exist in the queue, the speech synthesizer pauses a minimum amount of time equal to the sum of the current utterance,Äôs postUtteranceDelay and the next utterance,Äôs preUtteranceDelay.
(Read and Write property)

10.4.14 prefersAssistiveTechnologySettings as Boolean

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A Boolean that specifies whether assistive technology settings take precedence over the property values of this utterance.
Notes: If this property is true, but no assistive technology, such as VoiceOver, is on, the speech synthesizer uses the utterance property values.
(Read and Write property)

10.4.15 preUtteranceDelay as Double

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The amount of time the speech synthesizer pauses before speaking the utterance.
Notes: When multiple utterances exist in the queue, the speech synthesizer pauses a minimum amount of time equal to the sum of the current utterance,Äôs postUtteranceDelay and the next utterance,Äôs preUtteranceDelay.
(Read and Write property)

10.4.16 rate as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The rate the speech synthesizer uses when speaking the utterance.
Notes: The speech rate is a decimal representation within the range of AVSpeechUtteranceMinimumSpeechRate and AVSpeechUtteranceMaximumSpeechRate. Lower values correspond to slower speech, and higher values correspond to faster speech. The default value is AVSpeechUtteranceDefaultSpeechRate. Set this property before enqueing the utterance because setting it afterward has no effect.
(Read and Write property)

10.4.17 speechString as String

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: A string that contains the text for speech synthesis.
Notes: You can,Äôt change an utterance,Äôs text after initializaiton. If you want the speech synthesizer to speak different text, create a new utterance.
(Read only property)

10.4.18 voice as AVSpeechSynthesisVoiceMBS

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The voice the speech synthesizer uses when speaking the utterance.
Notes: If you don,Äôt specify a voice, the speech synthesizer uses the system,Äôs default voice to speak the utterance.
(Read and Write property)

10.4.19 volume as Single

Plugin Version: 24.1, Platforms: macOS, iOS, Targets: All.

Function: The volume the speech synthesizer uses when speaking the utterance.
Notes: Before enqueing the utterance, set this property to a value within the range of 0.0 for silent to 1.0 for loudest volume. The default value is 1.0. Setting this after enqueing the utterance has no effect.
(Read and Write property)

Chapter 11

List of Questions in the FAQ

Chapter 12

The FAQ

12.0.1 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)

12.0.2 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.

12.0.3 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:

- 12.0.4 How to catch delete key? 1179
- 12.0.5 How to convert cmyk to rgb? 1180
- 12.0.6 How to delete a folder? 1181
- 12.0.7 How to detect if CPU if 64bit processor? 1182
- 12.0.8 How to query variant type string for a variant? 1183
- 12.0.9 How to refresh a htmlviewer on Windows? 1184

12.0.4 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178

12.0.5 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178
- 12.0.4 How to catch delete key? 1179
- 12.0.6 How to delete a folder? 1181
- 12.0.7 How to detect if CPU if 64bit processor? 1182
- 12.0.8 How to query variant type string for a variant? 1183
- 12.0.9 How to refresh a htmlviewer on Windows? 1184

12.0.6 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178
- 12.0.4 How to catch delete key? 1179
- 12.0.5 How to convert cmyk to rgb? 1180
- 12.0.7 How to detect if CPU if 64bit processor? 1182
- 12.0.8 How to query variant type string for a variant? 1183
- 12.0.9 How to refresh a htmlviewer on Windows? 1184

12.0.7 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178
- 12.0.4 How to catch delete key? 1179
- 12.0.5 How to convert cmyk to rgb? 1180
- 12.0.6 How to delete a folder? 1181
- 12.0.8 How to query variant type string for a variant? 1183
- 12.0.9 How to refresh a htmlviewer on Windows? 1184

12.0.8 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178
- 12.0.4 How to catch delete key? 1179
- 12.0.5 How to convert cmyk to rgb? 1180
- 12.0.6 How to delete a folder? 1181
- 12.0.7 How to detect if CPU if 64bit processor? 1182
- 12.0.9 How to refresh a htmlviewer on Windows? 1184

12.0.9 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:

- 12.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1178
- 12.0.4 How to catch delete key? 1179
- 12.0.5 How to convert cmyk to rgb? 1180

- 12.0.6 How to delete a folder? 1181
- 12.0.7 How to detect if CPU if 64bit processor? 1182
- 12.0.8 How to query variant type string for a variant? 1183

12.0.10 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
```

12.0.11 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.

12.0.12 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
```

12.0.13 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.

12.0.14 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

12.0.15 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.

12.0.16 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.

12.0.17 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

12.0.18 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.

12.0.19 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.softLighting(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.

12.0.20 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.

12.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

12.0.22 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.

12.0.23 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.

12.0.24 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

12.0.25 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.

12.0.26 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.

12.0.27 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 "&amps;#nn;" (eg. ">" will become "&amps;#62;".). 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.

12.0.28 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

12.0.29 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.

12.0.30 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)

12.0.31 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.

12.0.32 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.

12.0.33 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.

12.0.34 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
```

12.0.35 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.

12.0.36 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.

12.0.37 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.

12.0.38 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.

12.0.39 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.

12.0.40 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.

12.0.41 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.

12.0.42 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.

12.0.43 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.

12.0.44 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
```

12.0.45 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.

12.0.46 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

12.0.47 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.

12.0.48 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.

12.0.49 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
```

12.0.50 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.

12.0.51 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.

12.0.52 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.

12.0.53 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.

12.0.54 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.

12.0.55 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.

12.0.56 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.

12.0.57 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.

12.0.58 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.

12.0.59 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.

12.0.60 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).

12.0.61 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.

12.0.62 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.

12.0.63 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>
```

12.0.64 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

12.0.65 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.

12.0.66 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)

12.0.67 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.

12.0.68 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.

12.0.69 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/ .

12.0.70 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.

12.0.71 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.

12.0.72 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
```

12.0.73 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.

12.0.74 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.

12.0.75 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.

12.0.76 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.

12.0.77 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.

12.0.78 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
```

12.0.79 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
```

12.0.80 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
```

12.0.81 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.

12.0.82 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.

12.0.83 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.

12.0.84 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

12.0.85 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.

12.0.86 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.

12.0.87 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.

12.0.88 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.

12.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

12.0.90 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.

12.0.91 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.

12.0.92 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.

12.0.93 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.

12.0.94 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)

12.0.95 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.

12.0.96 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".

12.0.97 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)
```

12.0.98 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.

12.0.99 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.

12.0.100 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.

12.0.101 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
```

12.0.102 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.

12.0.103 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.

12.0.104 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.

12.0.105 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

12.0.106 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.

12.0.107 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.

12.0.108 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.

12.0.109 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 &auml; to √§.

12.0.110 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
```

12.0.111 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
```

12.0.112 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.

12.0.113 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".

12.0.114 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/

12.0.115 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.

12.0.116 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
```

12.0.117 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
```

12.0.118 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.

12.0.119 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
```

12.0.120 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.

12.0.121 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.

12.0.122 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.

12.0.123 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

12.0.124 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.

12.0.125 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.

12.0.126 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" ...

12.0.127 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.

12.0.128 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
```

12.0.129 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
```

12.0.130 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.

12.0.131 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
```

12.0.132 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
```

12.0.133 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.

12.0.134 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.

12.0.135 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!

12.0.136 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
```

12.0.137 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.

12.0.138 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.

12.0.139 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.

12.0.140 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.

12.0.141 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.

12.0.142 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
```

12.0.143 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).

12.0.144 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.

12.0.145 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.

12.0.146 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.attributedStringWithAttachment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS method.

12.0.147 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
```

12.0.148 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
```

12.0.149 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.

12.0.150 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.

12.0.151 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.

12.0.152 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.

12.0.153 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
```

12.0.154 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.

12.0.155 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.

12.0.156 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.

12.0.157 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
```

12.0.158 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

12.0.159 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

12.0.160 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

12.0.161 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.

12.0.162 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?

12.0.163 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))
```

12.0.164 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.

12.0.165 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.

12.0.166 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)

12.0.167 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.

12.0.168 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.

12.0.169 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.

12.0.170 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
```

12.0.171 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.

12.0.172 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
```

12.0.173 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.

12.0.174 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.

12.0.175 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.

12.0.176 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)+”;”)
```

12.0.177 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

12.0.178 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.

12.0.179 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.

12.0.180 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.

12.0.181 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.

12.0.182 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.

12.0.183 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.

12.0.184 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
```

12.0.185 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.

12.0.186 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.

12.0.187 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
```

12.0.188 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.

12.0.189 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.

12.0.190 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.

12.0.191 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.

12.0.192 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.

12.0.193 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)

12.0.194 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.

12.0.195 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.

12.0.196 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
```

12.0.197 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
```

12.0.198 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.

12.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:
Example:

```
window1.ModifiedMBS=true
```

12.0.200 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.

12.0.201 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/SharedSupport/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
```

12.0.202 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.

12.0.203 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
```

12.0.204 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
```

12.0.205 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.

12.0.206 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.

12.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.
Example:

```
msgbox "This String contains ""quotes""."
```

12.0.208 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
```

12.0.209 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!

12.0.210 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.

12.0.211 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".

12.0.212 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.

12.0.213 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.

12.0.214 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.

12.0.215 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.

12.0.216 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.

12.0.217 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.

12.0.218 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)

12.0.219 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.

12.0.220 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

12.0.221 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.

12.0.222 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
```

12.0.223 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

12.0.224 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)

12.0.225 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?

12.0.226 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.

12.0.227 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.

12.0.228 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.

12.0.229 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.

12.0.230 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.

12.0.231 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.

12.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

12.0.233 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
```

12.0.234 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.

12.0.235 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.

12.0.236 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.

12.0.237 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.

12.0.238 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.

12.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

12.0.240 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.

12.0.241 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.

12.0.242 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.

12.0.243 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.

12.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

12.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

12.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

12.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

12.0.248 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.

12.0.249 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.

12.0.250 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.

12.0.251 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.

12.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:
http://www.wotsit.org

12.0.253 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

12.0.254 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.

12.0.255 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.

12.0.256 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.

12.0.257 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.

12.0.258 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.

12.0.259 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.

12.0.260 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.

12.0.261 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.

12.0.262 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.

12.0.263 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.

12.0.264 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

12.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

12.0.266 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„