MBS Win 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 Win Plugin

0.2   Content

Chapter 1

List of Topics

Chapter 2

List of all classes

Chapter 3

List of all controls

Chapter 4

List of all modules

Chapter 5

List of all global methods

Chapter 6

Cocoa Controls

6.1 class DesktopTextArea

6.1.1 class DesktopTextArea

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in textarea class in Xojo.
Example:

```
// make a PDF from a textarea in Cocoa Xojo target:

// find view
dim n as NSViewMBS = TextArea1.NSViewMBS
if n = nil then
MsgBox "Only in Cocoa!"
Return
end if

// make pdf data
dim s as string = n.dataWithPDFInsideRect(0,0,n.frame.Width, n.frame.Height)

// save
dim f as FolderItem = GetSaveFolderItem("", "test.pdf")

if f<>Nil then

dim b as BinaryStream = BinaryStream.Create(f, true)

b.Write s

end if
```

Notes: Requires Xojo 2021r3 or newer.

6.1.2   Methods

6.1.3   WinInsertImageMBS(data as string, Width as Integer, Height as Integer)

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Inserts image to file.
Example:

Dim pic As Picture = LogoMBS(500)
Dim data As String = pic.GetData(pic.SaveAsJPEG)

call TextArea1.WinInsertImageMBS(data, 500, 500)

Notes: Specify size in pixels and image data.
Image data works with JPEG and other data types.
Requires Windows 8 or newer.

6.1.4   WinShowFontPanelMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Shows font dialog.
Example:

If TextArea1.WinShowFontPanelMBS Then
// update controls as font may have been changed?

End If

Notes: You can make changes and apply them with OK button.
Returns true when OK is selected.

6.1.5 Properties

6.1.6 WinAutoCorrectionMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Enables auto correction for Textarea.
Example:

#If TargetWindows Then
TextArea1.WinAutoCorrectionMBS = True
#ElseIf TargetMacOS Then
TextArea1.NSTextViewMBS.AutomaticSpellingCorrectionEnabled = True
#EndIf

Notes: Only for Windows 8 and newer.
Uses current language from user.

See AutomaticSpellingCorrectionEnabled and ContinuousSpellCheckingEnabled properties in NSTextViewMBS for MacOS.
(Read and Write computed property)

6.1.7 WinAutoHorizontalScrollMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Whether to auto scroll horizontally.
Notes: Automatically scrolls text to the right by 10 characters when the user types a character at the end of the line. When the user presses the ENTER key, the control scrolls all text back to position zero.

This property controls an option of the RichEdit control used by Xojo for the Textarea control on Windows.
(Read and Write computed property)

6.1.8 WinAutoVerticalScrollMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Whether to auto scroll vertically.
Notes: Automatically scrolls text up one page when the user presses the ENTER key on the last line.

This property controls an option of the RichEdit control used by Xojo for the Textarea control on Windows.
(Read and Write computed property)

6.1.9   WinRTFDataMBS(SelectionOnly as boolean = false) as string

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Get or set the RTF data for the text area.
Example:

Dim rtfData As String

#If TargetWindows Then
rtfData = TextArea1.WinRTFDataMBS
#ElseIf TargetMacOS Then
rtfData = TextArea1.RTFDataMBS
#EndIf

Notes: Including images and probably faster as StyledText.RTFData.
(Read and Write computed property)

6.1.10   WinSelHasTextBackColorMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Whether a custom background color is in used for the selection.
Example:

textArea1.WinSelHasTextBackColorMBS = false

Notes: If false, standard background color is used.
(Read and Write computed property)

6.1.11   WinSelHasTextColorMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Whether a custom text color is in used for the selection.
Example:

textArea1.WinSelHasTextColorMBS = false

Notes: If false, standard text color is used.
(Read and Write computed property)

6.1.12   WinSelStrikeThroughMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Get or set StrikeThrough text style.
Example:

TextArea1.WinSelStrikeThroughMBS = True

Notes: See example project for how to do this for both Mac and Windows.
(Read and Write computed property)

6.1.13   WinSelSubScriptMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Get or set subscript text style.
Example:

TextArea1.WinSelSubScriptMBS = True

Notes: Subscript text has a lower base line and is smaller.
See example project for how to do this for both Mac and Windows.
(Read and Write computed property)

6.1.14   WinSelSuperScriptMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Get or set superscript text style.
Example:

TextArea1.WinSelSuperScriptMBS = True

Notes: Superscript text has a higher base line and is smaller.
See example project for how to do this for both Mac and Windows.

(Read and Write computed property)

6.1.15   WinSelTextBackColorMBS as Color

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: The current background text color.
Example:

textArea1.WinSelTextBackColorMBS = &cFF0000

Notes: If set also sets WinSelHasTextBackColorMBS to true internally.
(Read and Write computed property)

6.1.16   WinSelTextColorMBS as Color

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: The current foreground text color.
Example:

textArea1.WinSelTextColorMBS = &cFF0000

Notes: If set also sets WinSelHasTextColorMBS to true internally.
(Read and Write computed property)

6.1.17   WinSpellcheckingMBS as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Enables spellchecking for Textarea.
Example:

#If TargetWindows Then
TextArea1.WinSpellcheckingMBS = True
#ElseIf TargetMacOS Then
TextArea1.NSTextViewMBS.continuousSpellCheckingEnabled = True
#EndIf

Notes: Only for Windows 8 and newer.
Uses current language from user.

See AutomaticSpellingCorrectionEnabled and ContinuousSpellCheckingEnabled properties in NSTextViewMBS
for MacOS.
(Read and Write computed property)

6.2   class TextArea

6.2.1   class TextArea

Plugin Version: 9.7, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in textarea class in Xojo.
Example:

```
// make a PDF from a textarea in Cocoa Xojo target:

// find view
dim n as NSViewMBS = TextArea1.NSViewMBS
if n = nil then
MsgBox "Only in Cocoa!"
Return
end if

// make pdf data
dim s as string = n.dataWithPDFInsideRect(0,0,n.frame.Width, n.frame.Height)

// save
dim f as FolderItem = GetSaveFolderItem("", "test.pdf")

if f<>Nil then

dim b as BinaryStream = BinaryStream.Create(f, true)

b.Write s

end if
```

Notes: Requires RB 2009r4 or newer.

6.2.2   Methods

6.2.3   WinInsertImageMBS(data as string, Width as Integer, Height as Integer)

Plugin Version: 14.3, Platform: Windows, Targets: Desktop only.

Function: Inserts image to file.
Example:

```
Dim pic As Picture = LogoMBS(500)
Dim data As String = pic.GetData(pic.SaveAsJPEG)
```

call TextArea1.WinInsertImageMBS(data, 500, 500)

Notes: Specify size in pixels and image data.
Image data works with JPEG and other data types.
Requires Windows 8 or newer.
Blog Entries

- RTF functions in MBS Plugins

- MBS Xojo / Real Studio Plugins, version 14.3pr1

- New RTF functions coming for Windows

6.2.4 WinShowFontPanelMBS as Boolean

Plugin Version: 19.2, Platform: Windows, Targets: Desktop only.

Function: Shows font dialog.
Example:

If TextArea1.WinShowFontPanelMBS Then
// update controls as font may have been changed?

End If

Notes: You can make changes and apply them with OK button.
Returns true when OK is selected.
Blog Entries

- RTF functions in MBS Plugins

- MBS Xojo Plugins in version 19.2

- Windows Font Dialog

- MBS Xojo Plugins, version 19.2pr3

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 17.5, page 43: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes

6.2.5 Properties

6.2.6 WinAutoCorrectionMBS as Boolean

Plugin Version: 19.2, Platform: Windows, Targets: Desktop only.

Function: Enables auto correction for Textarea.
Example:

#If TargetWindows Then
TextArea1.WinAutoCorrectionMBS = True
#ElseIf TargetMacOS Then
TextArea1.NSTextViewMBS.AutomaticSpellingCorrectionEnabled = True
#EndIf

Notes: Only for Windows 8 and newer.
Uses current language from user.

See AutomaticSpellingCorrectionEnabled and ContinuousSpellCheckingEnabled properties in NSTextViewMBS for MacOS.
(Read and Write computed property)

6.2.7 WinAutoHorizontalScrollMBS as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Whether to auto scroll horizontally.
Notes: Automatically scrolls text to the right by 10 characters when the user types a character at the end of the line. When the user presses the ENTER key, the control scrolls all text back to position zero.

This property controls an option of the RichEdit control used by Xojo for the Textarea control on Windows.
(Read and Write computed property)

6.2.8 WinAutoVerticalScrollMBS as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Whether to auto scroll vertically.
Notes: Automatically scrolls text up one page when the user presses the ENTER key on the last line.

This property controls an option of the RichEdit control used by Xojo for the Textarea control on Windows.
(Read and Write computed property)

6.2.9 WinRTFDataMBS(SelectionOnly as boolean = false) as string

Plugin Version: 14.3, Platform: Windows, Targets: Desktop only.

Function: Get or set the RTF data for the text area.
Example:

Dim rtfData As String

#If TargetWindows Then
rtfData = TextArea1.WinRTFDataMBS
#ElseIf TargetMacOS Then
rtfData = TextArea1.RTFDataMBS
#EndIf

Notes: Including images and probably faster as StyledText.RTFData.

See also RTFDataMBS in StyledText, RTFDataMBS in TextArea and DesktopTextArea controls and RTF-Data in NSTextViewMBS class.
(Read and Write computed property)

6.2.10 WinSelHasTextBackColorMBS as Boolean

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: Whether a custom background color is in used for the selection.
Example:

textArea1.WinSelHasTextBackColorMBS = false

Notes: If false, standard background color is used.
(Read and Write computed property)

6.2.11 WinSelHasTextColorMBS as Boolean

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: Whether a custom text color is in used for the selection.
Example:

textArea1.WinSelHasTextColorMBS = false

Notes: If false, standard text color is used.
(Read and Write computed property)

6.2.12　WinSelStrikeThroughMBS as Boolean

Plugin Version: 17.4, Platform: Windows, Targets: Desktop only.

Function: Get or set StrikeThrough text style.
Example:

TextArea1.WinSelStrikeThroughMBS = True

Notes: See example project for how to do this for both Mac and Windows.
(Read and Write computed property)

6.2.13　WinSelSubScriptMBS as Boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: Get or set subscript text style.
Example:

TextArea1.WinSelSubScriptMBS = True

Notes: Subscript text has a lower base line and is smaller.
See example project for how to do this for both Mac and Windows.
(Read and Write computed property)

6.2.14　WinSelSuperScriptMBS as Boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: Get or set superscript text style.
Example:

TextArea1.WinSelSuperScriptMBS = True

Notes: Superscript text has a higher base line and is smaller.
See example project for how to do this for both Mac and Windows.
(Read and Write computed property)

6.2.15 WinSelTextBackColorMBS as Color

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: The current background text color.
Example:

textArea1.WinSelTextBackColorMBS = &cFF0000

Notes: If set also sets WinSelHasTextBackColorMBS to true internally.
(Read and Write computed property)

6.2.16 WinSelTextColorMBS as Color

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: The current foreground text color.
Example:

textArea1.WinSelTextColorMBS = &cFF0000

Notes: If set also sets WinSelHasTextColorMBS to true internally.
(Read and Write computed property)

6.2.17 WinSpellcheckingMBS as Boolean

Plugin Version: 18.2, Platform: Windows, Targets: Desktop only.

Function: Enables spellchecking for Textarea.
Example:

```
#If TargetWindows Then
TextArea1.WinSpellcheckingMBS = True
#ElseIf TargetMacOS Then
TextArea1.NSTextViewMBS.continuousSpellCheckingEnabled = True
```

#EndIf

Notes: Only for Windows 8 and newer.
Uses current language from user.

See AutomaticSpellingCorrectionEnabled and ContinuousSpellCheckingEnabled properties in NSTextViewMBS
for MacOS.
(Read and Write computed property)

Chapter 7

Controls

7.1   class Control

7.1.1   class Control

Plugin Version: 9.7, Platform: macOS, Targets: Desktop only.

Function: The built in Control class in Xojo.

7.1.2   Methods

7.1.3   WinClassNameMBS as string

Plugin Version: 14.3, Platform: Windows, Targets: Desktop only.

Function: Returns internal control class name for a Windows control.

7.2   class DesktopControl

7.2.1   class DesktopControl

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in Control class in Xojo.

7.2.2   Methods

7.2.3   WinClassNameMBS as string

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Returns internal control class name for a Windows control.

Chapter 8

Currency, Date and Time Format

8.1 class WinLocalizationMBS

8.1.1 class WinLocalizationMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for details about the current localization
Example:

dim l as WinLocalizationMBS
l=new WinLocalizationMBS

Notes: This class has a constructor to specify whether you want unicode or not.

You may write yourself functions to acquire the details you need in your application from the Windows and from the Mac classes.
Also you may consider to update your values as the user may change them while your application is running.
Blog Entries

- MBS Plugins 11.1 Release notes

- MBS REALbasic Plugins, version 11.1pr4

8.1.2 Methods

8.1.3 AbbreviatedDayName(index as Integer) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The localized name of the day.
Notes: Index from 0 to 6.
e.g. on a German Windows XP system: AbbreviatedDayName(0)="Mo"

8.1.4   AbbreviatedMonthName(index as Integer) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The localized month name.
Notes: Index from 0 to 12. 0=January, 11=December, 12=13th month if exists.
e.g. on a German Windows XP system: AbbreviatedMonthName(0)="Jan"

8.1.5   Constructor

Plugin Version: 5.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to get the values for the current locale.
Notes: There are four constructors:

- You can pass nothing to get the current locale.

- You can pass a Locale Identifier.

- You can pass a LanguageID and a SortID.

- You can pass a PrimaryLanguageID and a SubLanguageID combined with a SortID.

Use the LANG_* constants for the PrimaryLanguageID, the SUBLANG_* constants for the SubLanguageID and the SORT_* constants for the SortID.
See also:

8.1.6   Constructor(LanguageID as Integer, SortID as Integer)

Plugin Version: 9.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to query values for a given language.
Notes: There are four constructors:

- You can pass nothing to get the current locale.

- You can pass a Locale Identifier.

- You can pass a LanguageID and a SortID.

- You can pass a PrimaryLanguageID and a SubLanguageID combined with a SortID.

Use the LANG_* constants for the PrimaryLanguageID, the SUBLANG_* constants for the SubLanguageID and the SORT_* constants for the SortID.
See also:

8.1.7 Constructor(LCID as Integer)

Plugin Version: 9.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to query values for a given language.
Notes: There are four constructors:

- You can pass nothing to get the current locale.

- You can pass a Locale Identifier.

- You can pass a LanguageID and a SortID.

- You can pass a PrimaryLanguageID and a SubLanguageID combined with a SortID.

Use the LANG_* constants for the PrimaryLanguageID, the SUBLANG_* constants for the SubLanguageID and the SORT_* constants for the SortID.
See also:

8.1.8 Constructor(PrimaryLanguage as Integer, SubLanguage as Integer, SortID as Integer)

Plugin Version: 9.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to query values for a given language.
Notes: There are four constructors:

- You can pass nothing to get the current locale.

- You can pass a Locale Identifier.

- You can pass a LanguageID and a SortID.

- You can pass a PrimaryLanguageID and a SubLanguageID combined with a SortID.

Use the LANG_* constants for the PrimaryLanguageID, the SUBLANG_* constants for the SubLanguageID and the SORT_* constants for the SortID.
See also:

8.1.9   LongDayName(index as Integer) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The localized name of the day.
Notes: Index from 0 to 6.
e.g. on a German Windows XP system: LongDayName(0)="Montag"

8.1.10   LongMonthName(index as Integer) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The localized month name.
Notes: Index from 0 to 12. 0=January, 11=December, 12=13th month if exists.
e.g. on a German Windows XP system: LongMonthName(0)="Januar"

8.1.11   Properties

8.1.12   CalendarTypeSpecifier as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Current calendar type.
Notes: This type can be one of these values:

Value	Meaning
1	Gregorian (as in United States)
2	Gregorian (English strings always)
3	Era: Year of the Emperor (Japan)
4	Era: Year of Taiwan Region
5	Tangun Era (Korea)

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.13 CalendarTypeSpecifier2 as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Additional calendar types.
Notes: This can be a zero-separated list of one or more of these calendars type values:

Value	Meaning
0	No additional types valid
1	Gregorian (as in United States)
2	Gregorian (English strings always)
3	Era: Year of the Emperor (Japan)
4	Era: Year of Taiwan Region
5	Tangun Era (Korea)

e.g. on a German Windows XP system: ""
(Read only property)

8.1.14 CountryCode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Country code, based on international phone codes, also referred to as IBM country codes.
Notes: The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "49"

(Read only property)

8.1.15 CountryNameAbbreviated as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Abbreviated name of the country from the ISO Standard 3166.
Notes: e.g. on a German Windows XP system: "DEU"
(Read only property)

8.1.16 CountryNameAbbreviatedISO as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: ISO abbreviated country name.
Notes: e.g. on a German Windows XP system: "DE"
(Read only property)

8.1.17 CountryNameEnglish as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Full English name of the country.
Notes: This is always restricted to characters mappable into the ASCII 127-character subset.
e.g. on a German Windows XP system: "English"
(Read only property)

8.1.18 CountryNameLocalized as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Full localized name of the country.
Notes: e.g. on a German Windows XP system: "Deutschland"
(Read only property)

8.1.19 CountryNameNative as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Native name of the country.
Notes: e.g. on a German Windows XP system: "Deutschland"
(Read only property)

8.1.20   CurrencyDecimalSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) used as the monetary decimal separator.
Notes: e.g. on a German Windows XP system: ","
(Read only property)

8.1.21   CurrencyDigitsInternational as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of fractional digits for the international monetary format.
Notes: The maximum number of characters allowed for this string is 3.
e.g. on a German Windows XP system: "2"
(Read only property)

8.1.22   CurrencyDigitsLocalized as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of fractional digits for the local monetary format.
Notes: The maximum number of characters allowed for this string is 3.
e.g. on a German Windows XP system: "2"
(Read only property)

8.1.23   CurrencyGroupingMode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sizes for each group of monetary digits to the left of the decimal.
Notes: An explicit size is needed for each group; sizes are separated by semicolons. If the last value is zero, the preceding value is repeated. To group thousands, specify 3;0, for example.
e.g. on a German Windows XP system: "3;0"
(Read only property)

8.1.24 CurrencyNameEnglish as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The full English name of the currency associated with the locale.
Notes: e.g. on a German Windows XP system: "Euro"
Introduced with Windows 2000.
(Read only property)

8.1.25 CurrencyNameNative as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The native name of the currency associated with the locale.
Notes: e.g. on a German Windows XP system: "Euro"
Introduced with Windows 2000.
(Read only property)

8.1.26 CurrencyNegativeMode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Negative currency mode.
Notes: The maximum number of characters allowed for this string is 3. The mode can be one of the following values:

e.g. on a German Windows XP system: "8"
(Read only property)

8.1.27 CurrencyPositiveMode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Positive currency mode.
Notes: The maximum number of characters allowed for this string is 2. The mode can be one of the following values:

e.g. on a German Windows XP system: "3"
(Read only property)

Value	Example
0	($ 1.1)
1	–$ 1.1
2	$ –1.1
3	$ 1.1–
4	(1.1$)
5	–1.1$
6	1.1–$
7	1.1$ –
8	–1.1 $ (space before $)
9	–$ 1.1 (space after $)
10	1.1 $ –(space before $)
11	$ 1.1–(space after $)
12	$ –1.1 (space after $)
13	1.1–$ (space before $)
14	($ 1.1) (space after $)
15	(1.1 $) (space before $)

Value	Meaning
0	Prefix, no separation
1	Suffix, no separation
2	Prefix, 1-char. separation
3	Suffix, 1-char. separation

8.1.28 CurrencySymbolInternational as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Three characters of the international monetary symbol specified in ISO 4217, "Codes for the Representation of Currencies and Funds," followed by the character separating this string from the amount.
Example:

Dim WinLoc As WinLocalizationMBS
dim s as string

WinLoc=New WinLocalizationMBS

s=WinLoc.CurrencySymbolLocalized
msgbox str(lenb(s))+" "+str(len(s))+" "+s
s=WinLoc.CurrencySymbolInternational
msgbox str(lenb(s))+" "+str(len(s))+" "+s

Notes: e.g. on a German Windows XP system: "EUR"
(Read only property)

8.1.29   CurrencySymbolLocalized as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: String used as the local monetary symbol.
Example:

Dim WinLoc As WinLocalizationMBS
dim s as string

WinLoc=New WinLocalizationMBS

s=WinLoc.CurrencySymbolLocalized
msgbox str(lenb(s))+" "+str(len(s))+" "+s
s=WinLoc.CurrencySymbolInternational
msgbox str(lenb(s))+" "+str(len(s))+" "+s

Notes: e.g. on a German Windows XP system: ",Ç¨"
(Read only property)

8.1.30   CurrencyThousandSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) used as the monetary separator between groups of digits to the left of the decimal.
Notes: e.g. on a German Windows XP system: "."
(Read only property)

8.1.31   DateLeadingZerosDay as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for leading zeros in day fields.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	No leading zeros for days
1	Leading zeros for days

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.32 DateLeadingZerosMonth as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for leading zeros in month fields.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	No leading zeros for months
1	Leading zeros for months

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.33 DateLongFormatOrdering as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Long date format-ordering specifier.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	Month-Day-Year
1	Day-Month-Year
2	Year-Month-Day

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.34   DateLongFormatString as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Long date formatting string for this locale.
Notes: The string can consist of a combination of day, month, and year format pictures defined in the Day, Month, Year, and Era Format Pictures table in National Language Support Constants and any string of characters enclosed in single quotes. Characters in single quotes remain as given.
e.g. on a German Windows XP system: "dddd, d. MMMM yyyy"
(Read only property)

8.1.35   DateSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) for the date separator.
Notes: e.g. on a German Windows XP system: "."
(Read only property)

8.1.36   DateShortFormatOrdering as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Short date format-ordering specifier.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	Month-Day-Year
1	Day-Month-Year
2	Year-Month-Day

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.37 DateShortFormatString as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Short date formatting string for this locale.
Example:

dim w as new WinLocalizationMBS

MsgBox w.DateShortFormatString

Notes: The string can consist of a combination of day, month, and year format pictures defined in Day, Month, Year, and Era Format Pictures table in National Language Support Constants.
e.g. on a German Windows XP system: "dd.MM.yyyy"
(Read only property)

8.1.38 DateShortYearMonth as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Year/Month formatting string for the locale.
Notes: This string shows the proper format for a date string that contains only the year and the month.
e.g. on a German Windows XP system: "MMMM yyyy"
Introduced with Windows 2000.
(Read only property)

8.1.39 DecimalSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) used as the decimal separator.
Notes: e.g. on a German Windows XP system: ","
(Read only property)

8.1.40 DefaultCodePageANSI as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: American National Standards Institute (ANSI) code page associated with this locale.
Notes: The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "1252"

(Read only property)

8.1.41   DefaultCodePageEBCDIC as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Default EBCDIC code page associated with the locale.
Notes: The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "20273"
Introduced with Windows 2000.
(Read only property)

8.1.42   DefaultCodePageMac as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Original equipment manufacturer (OEM) code page associated with the country.
Notes: The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "10000"
(Read only property)

8.1.43   DefaultCodePageOEM as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Original equipment manufacturer (OEM) code page associated with the locale.
Notes: The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "850"
(Read only property)

8.1.44   DefaultCountryCode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Country code for the principal country in this locale.
Notes: This is provided so that partially specified locales can be completed with default values. The maximum number of characters allowed for this string is 6.
e.g. on a German Windows XP system: "49"
(Read only property)

8.1.45 DefaultLanguageID as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Language identifier for the principal language spoken in this locale.
Notes: This is provided so that partially specified locales can be completed with default values. The maximum number of characters allowed for this string is 5.
e.g. on a German Windows XP system: "0407"
(Read only property)

8.1.46 DigitGrouping as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sizes for each group of digits to the left of the decimal.
Notes: An explicit size is needed for each group; sizes are separated by semicolons. If the last value is zero, the preceding value is repeated. To group thousands, specify 3;0, for example.
e.g. on a German Windows XP system: "3;0"
(Read only property)

8.1.47 DigitSubstitution as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Unkown.
Notes: 0 = context, 1 = none, 2 = national
e.g. on a German Windows XP system: "1"
Introduced with Windows 2000.
(Read only property)

8.1.48 FirstDayOfWeek as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for the first day in a week.
Notes: Value from 0 to 6.
e.g. on a German Windows XP system: "0"
(Read only property)

8.1.49   FirstWeekOfYear as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for the first week of the year.
Notes: The specifier can be one of these values:

Value	Meaning
0	Week containing 1/1 is the first week of that year.
1	First full week following 1/1 is the first week of that year.
2	First week containing at least 4 days is the first week of that year.

e.g. on a German Windows XP system: "2"
(Read only property)

8.1.50   LanguageID as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Language identifier indicating the language.
Notes: The maximum number of characters allowed for this string is 5.
e.g. on a German Windows XP system: "0407"
(Read only property)

8.1.51   LanguageNameAbbreviated as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Abbreviated name of the language, created by taking the 2-letter language abbreviation from
the ISO Standard 639 and adding a third letter, as appropriate, to indicate the sublanguage.
Notes: e.g. on a German Windows XP system: "DEU"
(Read only property)

8.1.52   LanguageNameAbbreviatedISO as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: ISO abbreviated language name.
Notes: e.g. on a German Windows XP system: "de"
(Read only property)

8.1.53 LanguageNameEnglish as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Full English name of the language from the International Organization for Standardization (ISO) Standard 639.
Notes: This is always restricted to characters mappable into the ASCII 127-character subset.
e.g. on a German Windows XP system: "German"
(Read only property)

8.1.54 LanguageNameLocalized as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Full localized name of the language.
Notes: e.g. on a German Windows XP system: "Deutsch (Deutschland)"
(Read only property)

8.1.55 LanguageNameNativ as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Native name of the language.
Notes: e.g. on a German Windows XP system: "Deutsch"
(Read only property)

8.1.56 LeadingZeros as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for leading zeros in decimal fields.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	No leading zeros
1	Leading zeros

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.57   ListItemSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) used to separate list items.
Notes: For example, a comma is used in many locales.
e.g. on a German Windows XP system: ";"
(Read only property)

8.1.58   MeasureSystem as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: System of measurement.
Notes: This value is 0 if the metric system (Syst√©me International d'Unit√©s, or S.I.) is used and 1 if the U.S. system is used. The maximum number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "0"
(Read only property)

8.1.59   NativeASCII0to9 as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Native equivalents to ASCII 0 through 9.
Notes: e.g. on a German Windows XP system: "0123456789"
(Read only property)

8.1.60   NegativeNumberMode as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Negative number mode.
Notes: The mode can be one of these values:

Value	Meaning
0	(1.1)
1	−1.1
2	−1.1
3	1.1−
4	1.1 −

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.61   NegSepBySpace as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Separation of monetary symbol in a negative monetary value.
Notes: This value is 1 if the monetary symbol is separated by a space from the negative amount, 0 if it is not. The maximum number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "1"
(Read only property)

8.1.62   NegSymPrecedes as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Position of monetary symbol in a negative monetary value.
Notes: This value is 1 if the monetary symbol precedes the negative amount, 0 if it follows it. The maximum number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "0"
(Read only property)

8.1.63   NumberOfFraction as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of fractional digits.
Notes: The maximum number of characters allowed for this string is 3.
e.g. on a German Windows XP system: "2"
(Read only property)

8.1.64   Papersize as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Default paper size associated with the locale.
Notes: Possible values:

e.g. on a German Windows XP system: "9"
Introduced with Windows 2000.

1	Letter
5	Legal
8	DIN A3
9	DIN A4

(Read only property)

8.1.65   PosSepBySpace as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Separation of monetary symbol in a positive monetary value.
Notes: This value is 1 if the monetary symbol is separated by a space from a positive amount, 0 if it is not.
The maximum number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "1"
(Read only property)

8.1.66   PosSymPrecedes as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Position of monetary symbol in a positive monetary value.
Notes: This value is 1 if the monetary symbol precedes the positive amount, 0 if it follows it. The maximum
number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "0"
(Read only property)

8.1.67   SignNegative as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: String value for the negative sign.
Notes: e.g. on a German Windows XP system: "-"
(Read only property)

8.1.68   SignNegativePosition as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Formatting index for negative values.
Notes: This index uses the same values as SignPositivePosition. The maximum number of characters allowed for this string is 2.
e.g. on a German Windows XP system: "1"
(Read only property)

8.1.69 SignPositive as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: String value for the positive sign.
Notes: e.g. on a German Windows XP system: ""
(Read only property)

8.1.70 SignPositivePosition as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Formatting index for positive values.
Notes: The maximum number of characters allowed for this string is 2. The index can be one of the following values:

Value	Meaning
0	Parentheses surround the amount and the monetary symbol.
1	The sign string precedes the amount and the monetary symbol.
2	The sign string succeeds the amount and the monetary symbol.
3	The sign string immediately precedes the monetary symbol.
4	The sign string immediately succeeds the monetary symbol.

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.71 Sortname as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The full localized name of the sort for the given locale ID.
Notes: e.g. on a German Windows XP system: "W√rterbuch"
Introduced with Windows 2000.

(Read only property)

8.1.72 ThousandSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) used to separate groups of digits to the left of the decimal.
Notes: e.g. on a German Windows XP system: "."
(Read only property)

8.1.73 TimeAM as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: String for the AM designator.
Notes: e.g. on a German Windows XP system: ""
(Read only property)

8.1.74 TimeCenturyFormatSpecifier as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for full 4-digit century.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	Abbreviated 2-digit century
1	Full 4-digit century

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.75 TimeFormatSpecifier as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Time format specifier.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the

following values:

Value	Meaning
0	AM / PM 12-hour format
1	24-hour format

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.76 TimeLeadingZeros as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifier for leading zeros in time fields.
Notes: The maximum number of characters allowed for this string is 2. The specifier can be one of the following values:

Value	Meaning
0	No leading zeros for hours
1	Leading zeros for hours

e.g. on a German Windows XP system: "1"
(Read only property)

8.1.77 TimeMarkerPosition as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Unkown.
Notes: e.g. on a German Windows XP system: "0"
(Read only property)

8.1.78 TimePM as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: String for the PM designator.
Notes: e.g. on a German Windows XP system: ""

(Read only property)

8.1.79   TimeSeparator as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Character(s) for the time separator.
Notes: e.g. on a German Windows XP system: ":"
(Read only property)

8.1.80   TimeShortFormatString as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Time formatting strings for this locale.
Notes: The string can consist of a combination of the hour, minute, and second format pictures defined in the Hour, Minute, and Second Format Pictures table in National Language Support Constants.
e.g. on a German Windows XP system: "HH:mm:ss"
(Read only property)

8.1.81   Constants

Constants

Constant	Value	Description
LANG_AFRIKAANS	&h36	One of the constants for the LanguageID.
LANG_ALBANIAN	&h1c	One of the constants for the LanguageID.
LANG_ARABIC	1	One of the constants for the LanguageID.
LANG_ARMENIAN	&h2b	One of the constants for the LanguageID.
LANG_ASSAMESE	&h4d	One of the constants for the LanguageID.
LANG_AZERI	&h2c	One of the constants for the LanguageID.
LANG_BASQUE	&h2d	One of the constants for the LanguageID.
LANG_BELARUSIAN	&h23	One of the constants for the LanguageID.
LANG_BENGALI	&h45	One of the constants for the LanguageID.
LANG_BULGARIAN	2	One of the constants for the LanguageID.
LANG_CATALAN	3	One of the constants for the LanguageID.
LANG_CHINESE	4	One of the constants for the LanguageID.
LANG_CROATIAN	&h1a	One of the constants for the LanguageID.
LANG_CZECH	5	One of the constants for the LanguageID.
LANG_DANISH	6	One of the constants for the LanguageID.
LANG_DUTCH	&h13	One of the constants for the LanguageID.
LANG_ENGLISH	9	One of the constants for the LanguageID.
LANG_ESTONIAN	&h25	One of the constants for the LanguageID.
LANG_FAEROESE	&h38	One of the constants for the LanguageID.
LANG_FARSI	&h29	One of the constants for the LanguageID.
LANG_FINNISH	&h0b	One of the constants for the LanguageID.
LANG_FRENCH	&h0c	One of the constants for the LanguageID.
LANG_GEORGIAN	&h37	One of the constants for the LanguageID.
LANG_GERMAN	7	One of the constants for the LanguageID.
LANG_GREEK	8	One of the constants for the LanguageID.
LANG_GUJARATI	&h47	One of the constants for the LanguageID.
LANG_HEBREW	&h0d	One of the constants for the LanguageID.
LANG_HINDI	&h39	One of the constants for the LanguageID.
LANG_HUNGARIAN	&h0e	One of the constants for the LanguageID.
LANG_ICELANDIC	&h0f	One of the constants for the LanguageID.
LANG_INDONESIAN	&h21	One of the constants for the LanguageID.
LANG_ITALIAN	&h10	One of the constants for the LanguageID.
LANG_JAPANESE	&h11	One of the constants for the LanguageID.
LANG_KANNADA	&h4b	One of the constants for the LanguageID.
LANG_KASHMIRI	&h60	One of the constants for the LanguageID.
LANG_KAZAK	&h3f	One of the constants for the LanguageID.
LANG_KONKANI	&h57	One of the constants for the LanguageID.
LANG_KOREAN	&h12	One of the constants for the LanguageID.
LANG_LATVIAN	&h26	One of the constants for the LanguageID.
LANG_LITHUANIAN	&h27	One of the constants for the LanguageID.
LANG_MACEDONIAN	&h2f	One of the constants for the LanguageID.
LANG_MALAY	&h3e	One of the constants for the LanguageID.
LANG_MALAYALAM	&h4c	One of the constants for the LanguageID.
LANG_MANIPURI	&h58	One of the constants for the LanguageID.
LANG_MARATHI	&h4e	One of the constants for the LanguageID.
LANG_NEPALI	&h61	One of the constants for the LanguageID.
LANG_NEUTRAL	0	One of the constants for the LanguageID.
LANG_NORWEGIAN	&h14	One of the constants for the LanguageID.
LANG_ORIYA	&h48	One of the constants for the LanguageID.
LANG_POLISH	&h15	One of the constants for the LanguageID.
LANG_PORTUGUESE	&h16	One of the constants for the LanguageID.
LANG_PUNJABI	&h46	One of the constants for the LanguageID.
LANG_ROMANIAN	&h18	One of the constants for the LanguageID.
LANG_RUSSIAN	&h19	One of the constants for the LanguageID.
LANG_SANSKRIT	&h4f	One of the constants for the LanguageID.
LANG_SERBIAN	&h1a	One of the constants for the LanguageID.
LANG_SINDHI	&h59	One of the constants for the LanguageID.
LANG_SLOVAK	&h1b	One of the constants for the LanguageID.

Chapter 9

DDE

9.1 class DDEBinaryDataMBS

9.1.1 class DDEBinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Binary Data for DDEMBS conversation.
Notes: Be carefully: This objects for data received inside a DDEMBS Event are read only. Passing an object to a function normally destroys it.

9.1.2 Methods

9.1.3 Mem as memoryblock

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the data as a memoryblock.

9.1.4 size as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the size of this data object.

9.1.5   Str as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the data as a string.

9.1.6   Properties

9.1.7   Handle as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle for this data.
Notes: (Read and Write property)

9.1.8   Release as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: whether the destructor will release the data handle or not.
Notes: (Read and Write property)

9.2 class DDEContextInfoMBS

9.2.1 class DDEContextInfoMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Details for a connection.

9.2.2 Properties

9.2.3 Ansi as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: whether the other side works with Ansi strings (the normal ones).
Notes: Currently this DDEMBS Classes are only tested for ANSI Systems. And I'm not sure if Xojo itself does handle Unicode correctly on Windows.
(Read and Write property)

9.2.4 CountryID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The country ID for this connection.
Notes: Some Windows country constants:

(Read and Write property)

9.2.5 Flags as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Flags for this connection.
Notes: Currently not used by Windows.
(Read and Write property)

9.2.6 LangID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The language ID for this connection.
Notes: Some Windows language constants:

(Read and Write property)

9.2.7   Security as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: A user defined security code.
Notes: A security value, which may be anything you need.
Maybe a code to verify that it is your app or the encryption code?
(Read and Write property)

9.2.8   Unicode as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: whether the other side works with Unicode strings.
Notes: Currently this DDEMBS Classes are only tested for ANSI Systems. And I'm not sure if Xojo itself
does handle Unicode correctly on Windows.
(Read and Write property)

ALBANIA	355	Albania
ALGERIA	213	Algeria
ARGENTINA	54	Argentina
ARMENIA	374	Armenia
AUSTRALIA	61	Australia
AUSTRIA	43	Austria
AZERBAIJAN	994	Azerbaijan
BAHRAIN	973	Bahrain
BELARUS	375	Belarus
BELGIUM	32	Belgium
BELIZE	501	Belize
BOLIVIA	591	Bolivia
BRAZIL	55	Brazil
BRUNEI_DARUSSALAM	673	BruneiDarussalam
BULGARIA	359	Bulgaria
CANADA	2	Canada
CARIBBEAN	1	Caribbean
CHILE	56	Chile
COLOMBIA	57	Colombia
COSTA_RICA	506	CostaRica
CROATIA	385	Croatia
CZECH	420	CzechRepublic
DENMARK	45	Denmark
DOMINICAN_REPUBLIC	1	DominicanRepublic
ECUADOR	593	Ecuador
EGYPT	20	Egypt
EL_SALVADOR	503	ElSalvador
ESTONIA	372	Estonia
FAEROE_ISLANDS	298	FaeroeIslands
FINLAND	358	Finland
FRANCE	33	France
GEORGIA	995	Georgia
GERMANY	49	Germany
GREECE	30	Greece
GUATEMALA	502	Guatemala
HONDURAS	504	Honduras
HONG_KONG	852	HongKongS.A.R.,P.R.C.
HUNGARY	36	Hungary
ICELAND	354	Iceland
INDIA	91	India
INDONESIA	62	Indonesia
IRAN	981	Iran
IRAQ	964	Iraq
IRELAND	353	Ireland
ISRAEL	972	Israel
ITALY	39	Italy
JAMAICA	1	Jamaica
JAPAN	81	Japan
JORDAN	962	Jordan
KAZAKSTAN	7	Kazakstan
KENYA	254	Kenya
KUWAIT	965	Kuwait
LATVIA	371	Latvia
LEBANON	961	Lebanon
LIBYA	218	Libya
LIECHTENSTEIN	41	Liechtenstein
LITHUANIA	370	Lithuania
LUXEMBOURG	352	Luxembourg
MACAU	853	Macau
MACEDONIA	389	theFormerYugoslavRepublicofMacedonia
MALAYSIA	60	Malaysia
MEXICO	52	Mexico
MONACO	33	PrincipalityofMonaco
MOROCCO	212	Morocco
NETHERLANDS	31	Netherlands

NEUTRAL	&h00
AFRIKAANS	&h36
ALBANIAN	&h1c
ARABIC	&h01
ARMENIAN	&h2b
ASSAMESE	&h4d
AZERI	&h2c
BASQUE	&h2d
BELARUSIAN	&h23
BENGALI	&h45
BULGARIAN	&h02
CATALAN	&h03
CHINESE	&h04
CROATIAN	&h1a
CZECH	&h05
DANISH	&h06
DUTCH	&h13
ENGLISH	&h09
ESTONIAN	&h25
FAEROESE	&h38
FARSI	&h29
FINNISH	&h0b
FRENCH	&h0c
GEORGIAN	&h37
GERMAN	&h07
GREEK	&h08
GUJARATI	&h47
HEBREW	&h0d
HINDI	&h39
HUNGARIAN	&h0e
ICELANDIC	&h0f
INDONESIAN	&h21
ITALIAN	&h10
JAPANESE	&h11
KANNADA	&h4b
KASHMIRI	&h60
KAZAK	&h3f
KONKANI	&h57
KOREAN	&h12
LATVIAN	&h26
LITHUANIAN	&h27
MACEDONIAN	&h2f
MALAY	&h3e
MALAYALAM	&h4c
MANIPURI	&h58
MARATHI	&h4e
NEPALI	&h61
NORWEGIAN	&h14
ORIYA	&h48
POLISH	&h15
PORTUGUESE	&h16
PUNJABI	&h46
ROMANIAN	&h18
RUSSIAN	&h19
SANSKRIT	&h4f
SERBIAN	&h1a

9.3 class DDEMBS

9.3.1 class DDEMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives access to Windows DDE functions.
Notes: DDE on Windows is like AppleEvents on Mac OS, but not so comfortable.

9.3.2 Methods

9.3.3 clientTransaction(type as Integer,topic as DDEStringMBS) as DDEBinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a Transaction with the server.
Notes: Short version of ClientTransaction with datatype=CF_TEXT and data=nil.
See also:

- 9.3.4 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS) as DDEBinaryDataMBS   215

- 9.3.5 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as DDEBinaryDataMBS   215

9.3.4 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS) as DDEBinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a Transaction with the server.
Notes: Shortversion of ClientTransaction with datatype=CF_TEXT.
See also:

- 9.3.3 clientTransaction(type as Integer,topic as DDEStringMBS) as DDEBinaryDataMBS   215

- 9.3.5 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as DDEBinaryDataMBS   215

9.3.5 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as DDEBinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

216 CHAPTER 9. DDE

Function: Runs a Transaction with the server.
Example:

```
// List all Excel Topics

dim d as DDEMBS
dim s,topic as DDEStringMBS
dim g as DDEStringMBS
dim m as DDEBinaryDataMBS
dim t,enter,z as string
dim i,c as Integer

list.deleteAllRows
d=new DDEMBS
if d.InitClient then
s=d.newDDEString("Excel")
topic=d.newDDEString("System")
if topic<>nil and s<>nil and d.ConnectToServer(s,topic) then
g=d.newDDEString("Topics")
if g<>nil then
m=d.ClientTransaction(d.XTYP_REQUEST,g)
if m<>nil then
t=m.str
t=left(t,len(t)-1) // remove chr(0) at end
enter=chr(9)
c=countfields(t,Enter)
for i=1 to c
z=nthfield(t,enter,i)

list.addRow z
next
list.listindex=0
else
msgBox "Failed to transfer."
end if
else
msgBox "Fail to create second string."
end if
g=nil // must be released before close
else
msgBox "Fail to create connect."
end if
s=nil // must be released before close
d.close
else
msgBox "Fail to init for Client."
end if
```

Notes: Use this ClientTransaction if the application returns a value or ClientTransactionBoolean if it returns a boolean.
Data and Datatype are optional. If no datatype is set, the datatype is set to the value of CF_TEXT.
See also:

- 9.3.3 clientTransaction(type as Integer,topic as DDEStringMBS) as DDEBinaryDataMBS   215

- 9.3.4 clientTransaction(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS) as DDEBinaryDataMBS   215

9.3.6   clientTransactionBoolean(type as Integer,topic as DDEStringMBS) as Boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a Transaction with the server.
Notes: Short version of ClientTransactionBoolean with datatype=CF_TEXT and data=nil.
See also:

- 9.3.7 clientTransactionBoolean(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS) as Boolean   217

- 9.3.8 clientTransactionBoolean(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as Boolean   217

9.3.7   clientTransactionBoolean(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS) as Boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a Transaction with the server.
Notes: Short version of ClientTransactionBoolean with Datatype=CF_TEXT.
See also:

- 9.3.6 clientTransactionBoolean(type as Integer,topic as DDEStringMBS) as Boolean   217

- 9.3.8 clientTransactionBoolean(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as Boolean   217

9.3.8   clientTransactionBoolean(type as Integer,topic as DDEStringMBS, data as DDEBinaryDataMBS, datatype as Integer) as Boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a Transaction with the server.
Notes: Use this ClientTransaction if the application returns a value or ClientTransactionBoolean if it returns

a boolean.

Data and Datatype are optional. If no datatype is set, the datatype is set to the value of CF_TEXT.

See also:

9.3.9   close

Platform: Windows, Targets: Desktop, Console & Web.

Function: Closes the running connection.
Example:

```
dim d as DDEMBS
// work with dde object

if d<>nil then
d.close
d=nil
end if
```

Notes: RB seems to crash, if the object is not destroyed before application is quit.

9.3.10   ConnectToServer(appname as DDEStringMBS, topic as DDEStringMBS) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Connects to the application for the given topic.
Example:

```
dim a,b as DDEStringMBS
dim d as DDEMBS

d=new DDEMBS
if d.InitClient then
a=d.newDDEString("servicename")
b=d.newDDEString("topicname")
if a<>nil and b<>nil then
if d.ConnectToServer(a,b) then
msgBox "Connected."
return // ok, so leave before closing.
```

```
else
msgBox "Failed to connect."
end if
else
msgBox "Unable to make DDE Strings."
end if
d.close // failed

else
msgBox "Failed to initClient."
end if
```

Notes: Using nil for appname or topic you can try to connect to anyone who accepts.
Renamed in v4.3 to ConnectToServer from Connect for better RB 6 compatibility.

9.3.11   InitClient as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Starts DDE, but only for Client stuff.
Notes: With plugin version 6.1pr4 and newer you can have eight DDE objects in your application.
With older plugin versions you can only have one instance of the DDEMBS class at all!

9.3.12   InitServer as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Starts DDE for client and server stuff.
Notes: With plugin version 6.1pr4 and newer you can have eight DDE objects in your application.
With older plugin versions you can only have one instance of the DDEMBS class at all!

9.3.13   NewDDEBinaryData(name as DDEStringMBS,data as memoryblock,off-set as Integer,length as Integer,dataformat as Integer) as DDEBinary-DataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new binary data object.
See also:

- 9.3.14 NewDDEBinaryData(name as DDEStringMBS,data as string) as DDEBinaryDataMBS   220

9.3.14   NewDDEBinaryData(name as DDEStringMBS,data as string) as DDE-BinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new binary data object.
Notes: A short version of NewDDEBinaryData which takes the whole string.
See also:

9.3.15   NewDDEBinaryData(name as DDEStringMBS,data as string,offset as Integer,length as Integer) as DDEBinaryDataMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new binary data object.
Notes: The offset and length is optional.
See also:

9.3.16   NewDDEString(ansistring as string) as DDEStringMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new ANSI DDE string.
Example:

```
dim d as DDEMBS // your DDE object
dim a as DDEStringMBS
a=d.newDDEString("servicename")
```

Notes: ANSI is the normal string encoding on Windows.

9.3.17 NewDDEStringUnicode(unicodestring as string) as DDEStringMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new Unicode DDE string.
Example:

```
dim d as DDEMBS // your DDE object
dim a as DDEStringMBS
a=d.NewDDEStringUnicode("servicename")
```

Notes: Currently this DDE Classes are only tested for ANSI Systems. And I'm not sure if Xojo itself does handle Unicode correctly on Windows.

9.3.18 RegisterService(name as DDEStringMBS) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers a service name.
Example:

```
dim d as DDEMBS // your dde object
dim s as DDEStringMBS

if d<>nil then
s=d.newDDEString(editfield1.text)

if s<>nil then
if d.registerService(s) then
msgBox "Registered."
else
msgBox "Register fails."
end if
else
msgBox "Failed on NewDDEString"
end if
else
msgBox "No DDE object!?"
end if
```

9.3.19 UnRegisterService(name as DDEStringMBS) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Unregisters a service name.
Example:

dim d as DDEMBS // your dde object
dim s as DDEStringMBS

if d<>nil then
s=d.newDDEString(editfield1.text)

if s<>nil then
if d.UnRegisterService(s) then
msgBox "Unregistered."
else
msgBox "Unregister fails."
end if
else
msgBox "Failed on NewDDEString"
end if
else
msgBox "No DDE object!?"
end if

9.3.20   Properties

9.3.21   LastError as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read only property)

9.3.22   Timeout as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Timeout in milliseconds.
Example:

dim d as DDEMBS // your DDE object
d.timeout=10000 // 10 Seconds

Notes: Default is 1000.
(Read and Write property)

9.3.23 Events

9.3.24 AdviceData(topic as DDEStringMBS, item as DDEStringMBS, dataformat as Integer,data as DDEBinaryDataMBS) as Integer

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A dynamic data exchange (DDE) client receives the AdviceData event after establishing an advise loop with a server. This transaction informs the client that the value of the data item has changed.

Return Values:
A DDE callback function should return DDE_FACK if it processes this transaction, DDE_FBUSY if it is too busy to process this transaction, or DDE_FNOTPROCESSED if it rejects this transaction.

Remarks:
An application must copy the data associated with the DDEBinaryDataMBS object if the application must process the data after the callback function returns.

9.3.25 AdviceRequest(topic as DDEStringMBS, item as DDEStringMBS, dataformat as Integer,remaincount as Integer) as DDEBinaryDataMBS

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: The system sends the AdviceRequest event to a server after the server calls the DdePostAdvise function. This transaction informs the server that an advise transaction is outstanding on the specified topic name and item name pair and that data corresponding to the topic name and item name pair has changed.

Return Values:
The server should first call the NewDDEBinaryData function to create a data object that identifies the changed data and then return this object. The server should return nil if it is unable to complete the transaction.

9.3.26 AdviceStart(topic as DDEStringMBS, item as DDEStringMBS, dataformat as Integer) as Boolean

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A client uses this event to establish an advise loop with a server.

9.3.27   AdviceStop(topic as DDEStringMBS, item as DDEStringMBS, dataformat as Integer)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A client uses this event to end an advise loop with a server.

9.3.28   ConfirmConnect(topic as DDEStringMBS, service as DDEStringMBS, myself as Boolean)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: Your connection was accepted.

9.3.29   Connect(topic as DDEStringMBS, service as DDEStringMBS, myself as Boolean,info as DDEContextInfoMBS) as Boolean

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: Return true if you accept this connection.

9.3.30   Disconnect(myself as Boolean)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: The connection was closed.

9.3.31   Error(errorcode as Integer)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A dynamic data exchange (DDE) object receives the Error event when a critical error occurs.

Possible errors:

DMLERR_ADVACKTIMEOUT	&h4000	
DMLERR_BUSY	&h4001	
DMLERR_DATAACKTIMEOUT	&h4002	
DMLERR_DLL_NOT_INITIALIZED	&h4003	
DMLERR_DLL_USAGE	&h4004	
DMLERR_EXECACKTIMEOUT	&h4005	
DMLERR_INVALIDPARAMETER	&h4006	
DMLERR_LOW_MEMORY	&h4007	Memory is low; advise, poke, or execute data may be lost, or the system may fail.
DMLERR_MEMORY_ERROR	&h4008	
DMLERR_NOTPROCESSED	&h4009	
DMLERR_NO_CONV_ESTABLISHED	&h400a	
DMLERR_POKEACKTIMEOUT	&h400b	
DMLERR_POSTMSG_FAILED	&h400c	
DMLERR_REENTRANCY	&h400d	
DMLERR_SERVER_DIED	&h400e	
DMLERR_SYS_ERROR	&h400f	
DMLERR_UNADVACKTIMEOUT	&h4010	
DMLERR_UNFOUND_QUEUE_ID	&h4011	

9.3.32 Execute(topic as DDEStringMBS,data as DDEBinaryDataMBS) as Integer

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A dynamic data exchange (DDE) server object receives the Execute event when a client specifies XTYP_EXECUTE in the ClientTransaction function. A client uses this transaction to send a command string to the server.

Return Values:
A server should return DDE_FACK if it processes this transaction, DDE_FBUSY if it is too busy to process this transaction, or DDE_FNOTPROCESSED if it rejects this transaction.

The data property is only valid while inside the event.

9.3.33 Poke(topic as DDEStringMBS,item as DDEStringMBS,data as DDEBinaryDataMBS) as Integer

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: Return Values:
A server should return DDE_FACK if it processes this transaction, DDE_FBUSY if it is too busy to process this transaction, or DDE_FNOTPROCESSED if it rejects this transaction.

9.3.34 Register(application as DDEStringMBS,service as DDEStringMBS)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A service is registered. You may add it to your own list.

9.3.35 Request(topic as DDEStringMBS,item as DDEStringMBS,dataformat as Integer) as DDEBinaryDataMBS

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: The server should create a DDEBinaryDataMBS object that identifies the data and then return this object. The server should return nil if it is unable to complete the transaction. If the server returns nil, the client will receive a DDE_FNOTPROCESSED flag.

9.3.36 UnRegister(application as DDEStringMBS,service as DDEStringMBS)

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A service was unregistered.

9.3.37 WildConnect(topic as DDEStringMBS,service as DDEStringMBS, myself as boolean,info as DDEContextInfoMBS) as DDEStringPairListMBS

Platform: Windows, Targets: .

Function: One of the DDE events.
Notes: A connection is tried to all available servers. Return true if you accept.

9.3.38 Constants

Constants

Constant	Value	Description
CF_BITMAP	2	A constant for the format parameter. Specifies a Windows bitmap format
CF_DIB	8	A constant for the format parameter. Specifies the Windows Device Independent Bitmap (DIB) format.
CF_DIBV5	17	A constant for the format parameter. Specifies the Windows Device Independent Bitmap (DIB) format of Windows 2000 and newer.
CF_DIF	5	A constant for the format parameter. Specifies the Windows data interchange format.
CF_ENHMETAFILE	14	A constant for the format parameter. Specifies the Windows enhanced metafile format.
CF_HDROP	15	A constant for the format parameter. Specifies the Windows file drop format.
CF_LOCALE	16	A constant for the format parameter. Specifies the Windows locale format.
CF_METAFILEPICT	3	A constant for the format parameter. Specifies the Windows metafile format.
CF_OEMTEXT	7	A constant for the format parameter. Specifies the standard Windows original equipment manufacturer (OEM) text format.
CF_PALETTE	9	A constant for the format parameter. Specifies the Windows Palette format.
CF_PENDATA	10	A constant for the format parameter. Specifies the Windows pen data format.
CF_RIFF	11	A constant for the format parameter. Specifies the Resource Interchange File Format (RIFF) audio format.
CF_SYLK	4	A constant for the format parameter. Specifies the Windows symbolic link format.
CF_TEXT	1	A constant for the format parameter. Specifies the standard American National Standards Institute (ANSI) text format.
CF_TIFF	6	A constant for the format parameter. Specifies the Tagged Image File Format (TIFF).
CF_UNICODETEXT	13	A constant for the format parameter. Specifies the standard Windows Unicode text format.
CF_WAVE	12	A constant for the format parameter. Specifies the wave audio format.
DDE_FACK	&h8000	A constant for the return parameter of some events. Return DDE_FACK if everything was handled well.
DDE_FBUSY	&h4000	A constant for the return parameter of some events. Return DDE_FBUSY if your application is busy and can't currently handle the request.
DDE_FNOTPROCESSED	0	A constant for the return parameter of some events. Return DDE_FNOTPROCESSED if something went wrong will processing this event.
XTYP_EXECUTE	&h4050	A constant for the type parameter of ClientTransaction and ClientTransactionBoolean.
XTYP_POKE	&h4090	A constant for the type parameter of ClientTransaction and ClientTransactionBoolean.
XTYP_REQUEST	&h20B0	A constant for the type parameter of ClientTransaction and ClientTransactionBoolean.

9.4 class DDEStringMBS

9.4.1 class DDEStringMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A string for DDEMBS conversation.
Notes: Be carefully: This objects for data received inside a DDEMBS Event are read only. Passing an object to a function normally destroys it.

9.4.2 Methods

9.4.3 Len as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the length of this string.

9.4.4 Mem as memoryblock

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the content of this DDEMBS String as Xojo Memoryblock.

9.4.5 Str as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the content of this DDEMBS String as Xojo string.

9.4.6 Properties

9.4.7 Handle as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle for this data.
Notes: (Read and Write property)

9.4.8   Release as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: whether the destructor will release the data handle or not.
Notes: (Read and Write property)

9.5 class DDEStringPairListMBS

9.5.1 class DDEStringPairListMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A list of DDE string pairs.
Notes: Used for the wild connect event.

9.5.2 Methods

9.5.3 Append(item as DDEStringPairMBS)

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds another item to the list.

9.5.4 Count as Integer

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Counts the number of items in the list.

9.5.5 Item(index as Integer) as DDEStringPairMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an item from this list.
Notes: Returns nil on any error.
Index is from 0 to count-1.

9.6   class DDEStringPairMBS

9.6.1   class DDEStringPairMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A pair of topic and service DDEString objects.
Notes: Used for the wild connect event.

9.6.2   Properties

9.6.3   Service as DDEStringMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The service of this pair.
Notes: (Read and Write property)

9.6.4   Topic as DDEStringMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The topic of this pair.
Notes: (Read and Write property)

Chapter 10

DirectShow

10.1   class DirectShowAMCameraControlMBS

10.1.1   class DirectShowAMCameraControlMBS

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This interface provides local or remote control over a camera.
Example:

```realbasic
dim srcFilter as DirectShowBaseFilterMBS // your device
dim a as DirectShowAMCameraControlMBS = srcFilter.AMCameraControl

// query range for zoom
dim ZoomMin as integer
dim ZoomMax as integer
dim ZoomStep as integer
dim ZoomDefault as Integer
dim ZoomFlags as integer
a.GetRange(a.kPropertyZoom, ZoomMin, ZoomMax, ZoomStep, ZoomDefault, ZoomFlags)
dim ErrorCode as integer = a.Lasterror
dim ErrorText as string = a.LasterrorMessage

Break // check in debugger

// set a minimum value
a.Set(a.kPropertyZoom, (ZoomMax-ZoomMin)/2, 0)
ErrorCode = a.Lasterror
ErrorText = a.LasterrorMessage

Break // check in debugger
```

Notes: Applications can use this interface to control camera settings such as zoom, pan, aperture adjustment, or shutter speed. To obtain this interface, query the filter that controls the camera.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.1pr4

- MBS Real Studio Plugins, version 12.4pr9

10.1.2 Methods

10.1.3 Constructor

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.1.4 Get(PropertySelector as Integer, byref Value as Integer, byref Flags as Integer)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This method retrieves the current setting of a camera property.
Notes: PropertySelector: A long value that specifies the property to retrieve, see kProperty* constants.
Value: receives the value of the property.
Flags: Receives a member of the CameraControlFlags enumeration. The returned value indicates whether the setting is controlled manually or automatically.

Lasterror is set.

10.1.5 GetRange(PropertySelector as Integer, byref MinValue as Integer, byref MaxValue as Integer, byref SteppingDelta as Integer, byref DefaultValue as Integer, byref CapsFlags as Integer)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This method retrieves the range and default value of a specified camera property.
Example:

dim srcFilter as DirectShowBaseFilterMBS // your device
dim a as DirectShowAMCameraControlMBS = srcFilter.AMCameraControl

// query range for zoom
dim ZoomMin as integer
dim ZoomMax as integer
dim ZoomStep as integer
dim ZoomDefault as Integer
dim ZoomFlags as integer
a.GetRange(a.kPropertyZoom, ZoomMin, ZoomMax, ZoomStep, ZoomDefault, ZoomFlags)
dim ErrorCode as integer = a.Lasterror
dim ErrorText as string = a.LasterrorMessage

Break // check in debugger

Notes: PropertySelector: A long value that specifies the property to query, see kProperty* constants.
MinValue: Receives the minimum value of the property.
MaxValue: Receives the maximum value of the property.
SteppingDelta: Receives the step size for the property. The step size is the smallest increment by which the property can change.
DefaultValue: Receives the default value of the property.
CapsFlags: Receives an element of the CameraControlFlags enumeration, indicating whether the property is controlled automatically or manually.

Lasterror is set.

10.1.6 Set(PropertySelector as Integer, Value as Integer, Flags as Integer = 0)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This method sets a specified property on the camera.
Notes: PropertySelector: A long value that specifies the property to set, see kProperty* constants.
Value: A long value that specifes the new value of the property.
Flags: A long value that specifies the desired control setting. See kFlags* constants.

Lasterror is set.

10.1.7 ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string = "")

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

Function: Shows properties dialog.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.

Title: optional title for the dialog.

Sets lasterror property.
See also:

10.1.8   ShowPropertyDialog(parent as window = nil, x as integer = 0, y as integer = 0, title as string = "")

Plugin Version: 19.1, Platform: Windows, Targets: Desktop only.

Function: Shows properties dialog.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.1.9   Properties

10.1.10   Handle as Integer

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IAMCameraControl interface.
(Read and Write property)

10.1.11   Lasterror as Integer

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.1.12 LasterrorMessage as String

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.1.13 Constants

Flags

Constant	Value	Description
kFlagsAuto	1	The setting is controlled automatically.
kFlagsManual	2	The setting is controlled manually.

Properties

Constant	Value	Description
kPropertyExposure	4	Identifies the exposure setting, in log base 2 seconds. In other words, for values less than zero, the exposure time is 1/2n seconds, and for values zero or above, the exposure time is 2n seconds. For example: Value Seconds -3 1/8 -2 1/4 -1 1/2 0 1 1 2 2 4
kPropertyFocus	6	Specifies the camera's focus setting, as the distance to the optimally focused target, in millimeters. The range and default value are specific to the device.
kPropertyIris	5	Specifies the camera's iris setting, in units of fstop * 10.
kPropertyPan	0	Identifies the camera's pan setting, in degrees. Values range from –180 to +180, with the default set to zero. Positive values are clockwise from the origin (the camera rotates clockwise when viewed from above), and negative values are counterclockwise from the origin.
kPropertyRoll	2	Identifies the camera's roll setting, in degrees. Values range from –180 to +180, with the default set to zero. Positive values cause a clockwise rotation of the camera along the image-viewing axis, and negative values cause a counterclockwise rotation of the camera.
kPropertyTilt	1	Identifies the camera's tilt setting, in degrees. Values range from –180 to +180, with the default set to zero. Positive values point the imaging plane up, and negative values point the imaging plane down.
kPropertyZoom	3	Identifies the camera's zoom setting, in millimeters. Values range from 10 to 600, and the default is specific to the device.

10.2   class DirectShowAMCrossbarMBS

10.2.1   class DirectShowAMCrossbarMBS

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IAMCrossbar interface routes signals from an analog or digital source to a video capture filter.
Notes: This interface is implemented by the Analog Video Crossbar Filter. The Analog Video Crossbar filter is modeled after a general switching matrix, with n inputs and m outputs. For example, a video card might have two external connectors: a coaxial connector for TV, and an S-video input. These would be represented as input pins on the filter. To select one of the inputs, an application would use the IAMCrossbar interface to "route" an input pin to the filter's output pin, by calling the Route method.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd389171(v=vs.85).aspx
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo Plugins, version 19.1pr4

- MBS Xojo / Real Studio Plugins, version 14.1pr3

- MBS Real Studio Plugins, version 12.5pr1

10.2.2   Methods

10.2.3   BaseFilter as DirectShowBaseFilterMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the base filter for the crossbar.
Notes: See also:
http://msdn.microsoft.com/en-us/library/windows/desktop/dd390991(v=vs.85).aspx

10.2.4   CanRoute(OutputPinIndex as Integer, InputPinIndex as Integer) as boolean

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CanRoute method queries whether a specified input pin can be routed to a specified output pin.
Notes: OutputPinIndex: Specifies the index of the output pin.
InputPinIndex: Specifies the index of input pin.
Lasterror is set.

Returns true if two pins can be routed.

To route the pins, call the Route method. Output pins and input pins are both indexed from zero. To determine the number of output and input pins, call the getPinCounts method.

10.2.5   Constructor

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.2.6   GetCrossbarPinInfo(IsInputPin as boolean, PinIndex as Integer, byref PinIndexRelated as Integer, byref PhysicalType as Integer)

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The getCrossbarPinInfo method retrieves information about a specified pin.
Notes: IsInputPin: Specifies the direction of the pin. Use one of the following values.

Value	Meaning
True	Input pin
False	Output pin

PinIndex: Specifies the index of the pin.
PinIndexRelated: Variable that receives the index of the related pin, or –1 if no pin is related to this pin. The related pin is a pin on the same filter, with the same direction; it typically represents the same physical jack or connector. For example, a video tuner and an audio tuner might be related pins. Typically, if two pins are related, you should route them together.
PhysicalType: Variable that receives a member of the PhysicalConnectorType enumeration, indicating the pin's physical type.

Lasterror is set.
Output pins and input pins are both indexed from zero. To determine the number of output and input pins, call the getPinCounts method.

10.2.7   GetPinCounts(byref OutputPinCount as Integer, byref InputPinCount as Integer)

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The getPinCounts method retrieves the number of input and output pins on the crossbar filter.
Notes: OutputPinCount: Variable that receives the number of output pins.
InputPinCount: Variable that receives the number of input pins.

Lasterror is set.

The other IAMCrossbar methods take parameters that specify pins by index number. For these methods, output pins and input pins are both indexed from zero. Use the getPinCounts method to determine the upper bounds for each.

10.2.8   IsRoutedTo(InputPinIndex as Integer) as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IsRoutedTo method retrieves the input pin that is currently routed to the specified output pin.
Notes: OutputPinIndex: Specifies the index of the output pin.
InputPinIndex: Variable that receives the index of the input pin, or -1 if no input pin is routed to this output pin.

Output pins and input pins are both indexed from zero. To determine the number of output and input pins, call the getPinCounts method.
Lasterror is set.

10.2.9   PhysicalPinName(type as Integer) as string

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the name for a given type.
Notes: Just a convenience method to return english names for types.

10.2.10   Route(OutputPinIndex as Integer, InputPinIndex as Integer)

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Route method routes an input pin to an output pin.
Notes: OutputPinIndex: Specifies the index of the output pin.
InputPinIndex: Specifies the index of the input pin.

Lasterror is set.

Routing two pins causes the output pin to deliver data from that input pin. Only one input pin at a time can be routed to a given output pin.

Output pins and input pins are both indexed from zero. To determine the number of output and input pins, call the getPinCounts method.

10.2.11 ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string = "")

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

Function: Shows property dialog.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.2.12 ShowPropertyDialog(parent as window = nil, x as integer = 0, y as integer = 0, title as string = "")

Plugin Version: 19.1, Platform: Windows, Targets: Desktop only.

Function: Shows property dialog.
Example:

```
dim Capture as DirectShowCaptureGraphBuilderMBS // your capture graph
dim srcfilter as DirectShowBaseFilterMBS // your source filter

Dim AMCrossbar As DirectShowAMCrossbarMBS = srcfilter.AMCrossbar
If AMCrossbar <>Nil Then
AMCrossbar.ShowPropertyDialog
MsgBox "Result: "+Str(AMCrossbar.Lasterror)+": "+AMCrossbar.LasterrorMessage
Else
MsgBox "No AMCrossbar."
End If
```

Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.2.13   Properties

10.2.14   Handle as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IAMCrossbar interface.
(Read and Write property)

10.2.15   Lasterror as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.2.16   LasterrorMessage as String

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.2.17   Constants

Physical Type Constants

Constant	Value	Description
PhysConn_Audio_1394	4103	Specifies an IEEE 1394 pin for audio.
PhysConn_Audio_AESDigital	4099	Specifies an AES/EBU (Audio Engineering Society/European Broadcas Union) digital pin for audio.
PhysConn_Audio_AudioDecoder	4105	Specifies an audio decoder pin.
PhysConn_Audio_AUX	4102	Specifies an AUX pin for audio.
PhysConn_Audio_Line	4097	Specifies a line pin for audio.
PhysConn_Audio_Mic	4098	Specifies a microphone pin.
PhysConn_Audio_SCSI	4101	Specifies a SCSI pin for audio.
PhysConn_Audio_SPDIFDigital	4100	Specifies an S/PDIF (Sony/Philips Digital Interface Format) digital pin fo audio.
PhysConn_Audio_Tuner	4096	Specifies a tuner pin for audio.
PhysConn_Audio_USB	4104	Specifies a USB pin for audio.
PhysConn_Video_1394	10	Specifies an IEEE 1394 pin for video.
PhysConn_Video_AUX	9	Specifies an AUX (auxiliary) pin for video.
PhysConn_Video_Black	15	Not used.
PhysConn_Video_Composite	2	Specifies a composite pin for video.
PhysConn_Video_ParallelDigital	7	Specifies a parallel digital pin for video.
PhysConn_Video_RGB	4	Specifies an RGB pin for video.
PhysConn_Video_SCART	14	Specifies a SCART (Peritel) pin for video.
PhysConn_Video_SCSI	8	Specifies a SCSI (Small Computer System Interface) pin for video.
PhysConn_Video_SerialDigital	6	Specifies a serial digital pin for video.
PhysConn_Video_SVideo	3	Specifies an S-Video (Y/C video) pin.
PhysConn_Video_Tuner	1	Specifies a tuner pin for video.
PhysConn_Video_USB	11	Specifies a USB (Universal Serial Bus) pin for video.
PhysConn_Video_VideoDecoder	12	Specifies a video decoder pin.
PhysConn_Video_VideoEncoder	13	Specifies a video encoder pin.
PhysConn_Video_YRYBY	5	Specifies a YRYBY (Y, R–Y, B–Y) pin for video.

10.3   class DirectShowAMStreamConfigMBS

10.3.1   class DirectShowAMStreamConfigMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IAMStreamConfig interface sets the output format on certain capture and compression filters, for both audio and video.
Notes: Applications can use this interface to set format properties, such as the output dimensions and frame rate (for video) or the sample rate and number of channels (for audio).
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 21.1

- MBS Xojo Plugins, version 19.1pr4

- MBS Xojo / Real Studio Plugins, version 13.4pr7

10.3.2   Methods

10.3.3   AudioCaps as DirectShowAudioStreamConfigCapsMBS()

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the audio capabilities.
Notes: Lasterror is set.
Returns empty array for video streams.

10.3.4   Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.3.5   MediaTypes as DirectShowMediaTypeMBS()

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries media types.
Notes: Lasterror is set.
see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd319787(v=vs.85).aspx

10.3.6 NumberOfCapabilities as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the number of format capabilities that this pin supports.
Notes: Lasterror is set.

10.3.7 ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string = "")

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

Function: Shows property dialog.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.3.8 ShowPropertyDialog(parent as window = nil, x as integer = 0, y as integer = 0, title as string = "")

Plugin Version: 19.1, Platform: Windows, Targets: Desktop only.

Function: Shows property dialog.
Example:

```
dim Capture as DirectShowCaptureGraphBuilderMBS // your capture graph
dim srcfilter as DirectShowBaseFilterMBS // your source filter

Dim AMStreamConfig2 As DirectShowAMStreamConfigMBS = capture.GetStreamConfig(False, srcfilter)
If AMStreamConfig2 <>Nil Then
AMStreamConfig2.ShowPropertyDialog

MsgBox "Result: "+Str(AMStreamConfig2.Lasterror)+": "+AMStreamConfig2.LasterrorMessage
Else
MsgBox "No AMStreamConfig with srcfilter."
End If
```

Notes: Parent: the parent window for the dialog.

X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.3.9   VideoCaps as DirectShowVideoStreamConfigCapsMBS()

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the video capabilities.
Notes: Lasterror is set.
Returns empty array for audio streams.

10.3.10   Properties

10.3.11   Handle as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

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

10.3.12   Lasterror as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.3.13   LasterrorMessage as String

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.3.14 Format as DirectShowMediaTypeMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set the preferred output format.
Notes: Lasterror is set.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd319785(v=vs.85).aspx
http://msdn.microsoft.com/en-us/library/windows/desktop/dd319788(v=vs.85).aspx
(Read and Write computed property)

10.4   class DirectShowAMVideoCompressionMBS

10.4.1   class DirectShowAMVideoCompressionMBS

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IAMVideoCompression interface sets and retrieves video compression properties.
Notes: It is supported by some video compression filters, and also by some video capture filters that output compressed video. Filters that support this interface expose it through their output pins.
An application can use this interface to control how video is compressed, including characteristics such as the key-frame rate or the compression quality.
A filter that supports this interface might not support every method. Use the Capabilities property to determine which methods the filter supports.
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 19.1

- MBS Xojo Plugins, version 19.1pr4

- MBS Xojo Plugins, version 19.1pr3

Xojo Developer Magazine

- 17.5, page 43: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes

10.4.2   Methods

10.4.3   BaseFilter as DirectShowBaseFilterMBS

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the base filter.
Notes: Lasterror and LastErrorMessage are set.

10.4.4   Constructor

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.4.5 OverrideFrameSize(FrameNumber as Integer, Size as Integer)

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The frame size of a specified frame.
Notes: The first frame that the filter delivers is numbered zero.
If the filter supports this method, the CompressionCaps_CanCrunch flag is in the Capabilities property. However, this flag can also indicate that the filter supports setting the bit rate, so it does not guarantee that the OverrideFrameSize method is supported.

Lasterror and LastErrorMessage are set.

10.4.6 OverrideKeyFrame(FrameNumber as Integer)

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The filter to compress a particular frame as a key frame.
Notes: The first frame that the filter delivers is numbered zero.
If the filter supports this method, you can use it to override the normal key-frame distribution for a particular frame. After the filter creates a key frame, it might reset its count to determine when the next key frame should occur. For example, if the key-frame rate is 10, and an application uses this method to force frame 5 as a key frame, the filter might wait another 10 frames (until frame 15) before it creates the next key frame.

Lasterror and LastErrorMessage are set.

10.4.7 ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string = "")

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

Function: Shows property dialog.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.4.8   ShowPropertyDialog(parent as window = nil, x as integer = 0, y as integer = 0, title as string = "")

Plugin Version: 19.1, Platform: Windows, Targets: Desktop only.

Function: Shows property dialog.
Example:

dim Capture as DirectShowCaptureGraphBuilderMBS // your capture graph
dim encoder as DirectShowBaseFilterMBS // your encoder filter

Dim AMVideoCompression As DirectShowAMVideoCompressionMBS = encoder.AMVideoCompression
If AMVideoCompression <>Nil Then
AMVideoCompression.ShowPropertyDialog
MsgBox "Result: "+Str(AMVideoCompression.Lasterror)+": "+AMVideoCompression.LasterrorMessage
Else
MsgBox "No AMVideoCompression."
end if

Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

- 10.4.7 ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string
  = "")   249

10.4.9   Properties

10.4.10   Capabilities as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives the compression capabilities, as a bitwise combination of zero or more CompressionCaps flags.
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.11   DefaultKeyFrameRate as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives the default key-frame rate.
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.12   DefaultPFramesPerKey as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives the default rate of predicted (P) frames per key frame.
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.13   DefaultQuality as Double

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives the default quality.
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.14   Description as String

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives a description string, such as "My Video Compressor."
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.15   Handle as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

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

10.4.16   KeyFrameRate as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The key-frame rate.
Notes: If the value is negative, the filter will use the default key-frame rate. If the value is zero, only the first frame will be a key frame.

To determine if the filter supports this method, call the Capabilities property and check for the CompressionCaps_CanKeyFrame flag.

Lasterror and LastErrorMessage are set.
(Read and Write property)

10.4.17 Lasterror as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.4.18 LasterrorMessage as String

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.4.19 PFramesPerKeyFrame as Integer

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The rate of predicted (P) frames per key frame.
Notes: If the value is negative, the filter will use the default rate.
To determine if the filter supports this method, check for the CompressionCaps_CanBFrame flag in the Capabilities property.

Lasterror and LastErrorMessage are set.
(Read and Write property)

10.4.20 Quality as Double

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The compression quality.
Notes: Specifies the quality as a value between 0.0 and 1.0, where 1.0 indicates the best quality and 0.0 indicates the worst quality. If the value is negative, the filter will use the default quality.
To determine if the filter supports this method, check for the CompressionCaps_CanQuality flag in the Capabilities property.

Lasterror and LastErrorMessage are set.
(Read and Write property)

10.4.21 Version as String

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Receives a version string, such as "Version 2.1.0."
Notes: Lasterror and LastErrorMessage are set.
(Read only property)

10.4.22 WindowSize as UInt64

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of frames over which the compressor must maintain an average data rate.
Notes: For example, assuming a data rate of 100K/sec and a frame rate of 10 frames per second, if the window size is 1, then every frame will be 10K or less. If the window size is 5, then every five consecutive frames must average 10K per frame, but individual frames may exceed this size.
The window size, expressed as a number of frames.

Lasterror and LastErrorMessage are set.
(Read and Write property)

10.4.23 Constants

Compression Capabilities

Constant	Value	Description
CompressionCapsCanBFrame	&h08	Video compressor supports the PFramesPerKeyFrame properties.
CompressionCapsCanCrunch	&h02	Video compressor can compress video to a specified data rate. If the compressor has this capability then the output pins media type will contain the data rate in the VIDEOINFOHEADER structure's BitRate member. The only way to set the data rate is to set BitRate.
CompressionCapsCanKeyFrame	&h04	Video compressor supports the KeyFrameRate properties.
CompressionCapsCanQuality	&h01	Video compressor supports the Quality property.
CompressionCapsCanWindow	&h10	Video compressor supports the WindowSize properties.

10.5   class DirectShowAMVideoControlMBS

10.5.1   class DirectShowAMVideoControlMBS

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IAMVideoControl interface controls certain video capture operations such as enumerating available frame rates and image orientation.
Notes: For Windows Driver Model (WDM) devices, the WDM Video Capture Filter automatically exposes this interface if the WDM driver supports the PROPSETID_VIDCAP_VIDEOCONTROL property set. For more information, see the Windows Driver Kit (WDK) documentation.
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 21.1

- MBS Xojo Plugins, version 21.1pr1

Xojo Developer Magazine

- 19.3, page 10: News

10.5.2   Methods

10.5.3   Caps(pin as DirectShowPinMBS) as Integer

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetCaps method retrieves the capabilities of the underlying hardware.
Notes: Result is a combination of the flags from the VideoControl Flags, which specify the video control mode.

Possible capabilities include one or more of the following: flipping the picture horizontally, flipping the picture vertically, enabling external triggers, and simulating external triggers.

10.5.4   Constructor

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.5.5   CurrentActualFrameRate(pin as DirectShowPinMBS) as Int64

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetCurrentActualFrameRate method retrieves the actual frame rate, expressed as a frame duration in 100-nanosecond units. USB (Universal Serial Bus) and IEEE 1394 cameras may provide lower frame rates than requested because of bandwidth availability. This is only available during video streaming.
Notes: Pin: the pin to retrieve the frame rate from.

Lasterror is set.

10.5.6   FrameRateList(pin as DirectShowPinMBS, Index as Integer, Width as Integer, Height as Integer) as Int64()

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetFrameRateList method retrieves a list of available frame rates.
Notes: Pin: The pin to query for the list of frame rates.
Index: Index of the format to query for frame rates. This index corresponds to the order in which formats are enumerated by DirectShowVideoStreamConfigCapsMBS structures returned by DirectShowAMStreamConfigMBS.NumberOfCapabilities) minus one.
Width and Height: Frame image size (width and height) in pixels.

Returns an array of frame rates in 100-nanosecond units.

Lasterror is set.

10.5.7   MaxAvailableFrameRate(pin as DirectShowPinMBS, Index as Integer, Width as Integer, Height as Integer) as Int64

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetMaxAvailableFrameRate method retrieves the maximum frame rate currently available, based on bus bandwidth usage for connections, such as USB and IEEE 1394, where the maximum frame rate may be limited by bandwidth availability.
Notes: Pin: The pin to retrieve the maximum frame rate from.
Index: Index of the format to query for maximum frame rate. This index corresponds to the order in which formats are enumerated by GetStreamCaps. The value must range between zero and the number of supported DirectShowVideoStreamConfigCapsMBS returned by DirectShowAMStreamConfigMBS.NumberOfCapabilities) minus one.
Width and Height: Frame image size (width and height) in pixels.
MaxAvailableFrameRate: Pointer to the maximum available frame rate. The frame rate is expressed as frame duration in 100-nanosecond units.

Lasterror is set.

10.5.8   Mode(pin as DirectShowPinMBS) as Integer

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetMode method retrieves the video control mode of operation.
Notes: Possible modes of operation include one or more of the following: flipping the picture horizontally, flipping the picture vertically, enabling external triggers, and simulating external triggers.

Lasterror is set.

10.5.9   SetMode(pin as DirectShowPinMBS, mode as Integer)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetMode method sets the video control mode of operation.
Example:

dim pin as DirectShowPinMBS // your till image pin
dim filter as DirectShowBaseFilterMBS

dim vc as DirectShowAMVideoControlMBS = filter.AMVideoControl
vc.SetMode(pin, vc.FlagExternalTriggerEnable + vc.FlagTrigger)

Notes: Value specifying a combination of the flags from the VideoControl Flags to set the video control mode.

Possible modes of operation include one or more of the following: flipping the picture horizontally, flipping the picture vertically, enabling external triggers, and simulating external triggers.

Lasterror is set.

10.5.10   Properties

10.5.11   Handle as Integer

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IAMVideoControl interface.
(Read and Write property)

10.5.12   Lasterror as Integer

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable
error message.
(Read and Write property)

10.5.13   LasterrorMessage as String

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.5.14   Constants

Capabilities Flags

Constant	Value	Description
FlagExternalTriggerEnable	2	Sets up a stream to capture a trigger from an external source, for example, a push button on a camera. Buffers can be queued to the driver but will not be passed up from the WDM capture driver (for compression, display, or writing to a file) until the external event happens.
FlagFlipHorizontal	0	Specifies that the picture is flipped horizontally.
FlagFlipVertical	1	Specifies that the picture is flipped vertically.
FlagTrigger	3	In software, simulates an external trigger when the stream has the VideoControlFlag_ExternalTriggerEnable flag set.

10.6 class DirectShowAMVideoProcAmpMBS

10.6.1 class DirectShowAMVideoProcAmpMBS

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The AMVideoProcAmp interface adjusts the qualities of an incoming video signal, such as brightness, contrast, hue, saturation, gamma, and sharpness.
Notes: The WDM Video Capture filter exposes this interface if the hardware supports image adjustment. 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 12.4pr9

10.6.2 Methods

10.6.3 Constructor

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.6.4 Get(PropertySelector as Integer, byref Value as Integer, byref Flags as Integer)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets video quality for a specified property.
Notes: PropertySelector: Specifies the property to retrieve, as a value from the kProperty* constants.
Value: Receives the value of the property.
Flags: Receives the flags. The returned value indicates whether the setting is controlled manually or automatically.
Lasterror is set.

10.6.5 GetRange(PropertySelector as Integer, byref MinValue as Integer, byref MaxValue as Integer, byref SteppingDelta as Integer, byref DefaultValue as Integer, byref CapsFlags as Integer)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the minimum, maximum, and default values for setting properties.
Notes: PropertySelector: Specifies the property to retrieve, as a value from the kProperty* constants.
MinValue: Receives the minimum value of the property.

MaxValue: Receives the maximum value of the property.
SteppingDelta: Receives the step size for the property. The step size is the smallest increment by which the property can change.
DefaultValue: Receives the default value of the property.
CapsFlags: Receives a member of the VideoProcAmpFlags enumeration, indicating whether the property is controlled automatically or manually.
Lasterror is set.

10.6.6   Set(PropertySelector as Integer, Value as Integer, Flags as Integer = 0)

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets video quality for a specified property.
Notes: PropertySelector: Specifies the property to retrieve, as a value from the kProperty* constants.
Value: The new value of the property.
Flags: The desired control setting. See kFlags* constants.

Lasterror is set.

10.6.7   Properties

10.6.8   Handle as Integer

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IAMVideoProcAmp interface.
(Read and Write property)

10.6.9   Lasterror as Integer

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.6.10   LasterrorMessage as String

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.6.11   Constants

Flags

Constant	Value	Description
kFlagsAuto	1	The setting is controlled automatically.
kFlagsManual	2	The setting is controlled manually.

Properties

Constant	Value	Description
kPropertyBacklightCompensation	8	Specifies the backlight compensation setting. Possible values are 0 (off) and (on).
kPropertyBrightness	0	Specifies the brightness, also called the black level. For NTSC, the value i expressed in IRE units * 100. For non-NTSC sources, the units are arbitrary with zero representing blanking and 10,000 representing pure white. Value range from –10,000 to 10,000.
kPropertyColorEnable	6	Specifies the color enable setting. The possible values are 0 (off) and 1 (on).
kPropertyContrast	1	Specifies the contrast, expressed as gain factor * 100. Values range from zer to 10,000.
kPropertyGain	9	Specifies the gain adjustment. Zero is normal. Positive values are brighter an negative values are darker. The range of values depends on the device.
kPropertyGamma	5	Specifies the gamma, as gamma * 100. Values range from 1 to 500.
kPropertyHue	2	Specifies the hue, in degrees * 100. Values range from -180,000 to 180,000 (-18 to +180 degrees).
kPropertySaturation	3	Specifies the saturation. Values range from 0 to 10,000.
kPropertySharpness	4	Specifies the sharpness. Values range from 0 to 100.
kPropertyWhiteBalance	7	Specifies the white balance, as a color temperature in degrees Kelvin. Th range of values depends on the device.

10.7   class DirectShowAudioStreamConfigCapsMBS

10.7.1   class DirectShowAudioStreamConfigCapsMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes a range of audio formats.
Notes: Audio compression and capture filters use this structure to describe the formats they can produce.
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 13.4pr7

10.7.2   Methods

10.7.3   Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.7.4   Properties

10.7.5   BitsPerSampleGranularity as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Granularity of the bits per sample.
Notes: For example, the filter might specify 8 bits per sample through 32 bits per sample, in steps of 8.
(Read only property)

10.7.6   ChannelsGranularity as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Granularity of the channels.
Notes: For example, the filter might specify channels 2 through 4, in steps of 2.
(Read only property)

10.7.7 MaximumBitsPerSample as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum sample frequency.
Notes: (Read only property)

10.7.8 MaximumChannels as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum number of channels.
Notes: (Read only property)

10.7.9 MaximumSampleFrequency as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum sample frequency.
Notes: (Read only property)

10.7.10 MinimumBitsPerSample as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum bits per sample.
Notes: (Read only property)

10.7.11 MinimumChannels as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum number of channels.
Notes: (Read only property)

10.7.12 MinimumSampleFrequency as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum sample frequency.
Notes: (Read only property)

10.7.13   SampleFrequencyGranularity as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Granularity of the frequency.
Notes: For example, the filter might specify 11025 Hz to 44100 Hz, in steps of 11025 Hz.
(Read only property)

10.8 class DirectShowBaseFilterMBS

10.8.1 class DirectShowBaseFilterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The BaseFilter interface is the primary interface for DirectShow filters.
Notes: All DirectShow filters must expose this interface. The Filter Graph Manager uses this interface to control filters. Applications can use this interface to enumerate pins and query for filter information, but should not use it to change the state of a filter. Instead, use the MediaControl interface on the Filter Graph Manager.
Subclass of the DirectShowMediaFilterMBS class.
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 21.1

- MBS Xojo Plugins, version 19.1pr4

10.8.2 Methods

10.8.3 AMCameraControl as DirectShowAMCameraControlMBS

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries camera control object.
Example:

dim srcfilter as DirectShowBaseFilterMBS // your basefilter
dim value, flags as Integer
dim c as DirectShowAMCameraControlMBS = srcfilter.AMCameraControl
if c<>nil then

c.Get(c.kPropertyZoom, value, flags)
if c.Lasterror = 0 then
MsgBox "Zoom: "+str(value)
else
MsgBox "Zoom: "+c.LasterrorMessage
end if

c.Get(c.kPropertyFocus, value, flags)
if c.Lasterror = 0 then
MsgBox "Focus: "+str(value)
else
MsgBox "Focus: "+c.LasterrorMessage
end if

c.Get(c.kPropertyExposure, value, flags)
if c.Lasterror = 0 then
MsgBox "Exposure: "+str(value)
else
MsgBox "Exposure: "+c.LasterrorMessage
end if

end if

Notes: Returns object when this filter represents a device supporting this interface.
Else returns nil on any error.
Also sets lasterror property.

10.8.4   AMCrossbar as DirectShowAMCrossbarMBS

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the IAMCrossbar interface for this filter.
Notes: Returns nil if no such interface exists.

10.8.5   AMVideoCompression as DirectShowAMVideoCompressionMBS

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries video compression settings for the filter.

10.8.6   AMVideoControl as DirectShowAMVideoControlMBS

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the video controls for this filter.
Notes: Should work for capture filters.
Lasterror is set.

10.8.7   AMVideoProcAmp as DirectShowAMVideoProcAmpMBS

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries video properties object.
Example:

dim srcfilter as DirectShowBaseFilterMBS // your basefilter
dim value, flags as Integer
dim v as DirectShowAMVideoProcAmpMBS = srcfilter.AMVideoProcAmp
if v<>nil then
v.Get(v.kPropertyHue, value, flags)
if v.Lasterror = 0 then
MsgBox "Hue: "+str(value)
else
MsgBox "Hue: "+v.LasterrorMessage
end if

v.Get(v.kPropertyBrightness, value, flags)
if v.Lasterror = 0 then
MsgBox "Brightness: "+str(value)
else
MsgBox "Brightness: "+v.LasterrorMessage
end if

end if

Notes: Returns object when this filter represents a device supporting this interface.
Else returns nil on any error.
Also sets lasterror property.

10.8.8   ConfigAviMux as DirectShowConfigAviMuxMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the config avimux object.
Notes: Lasterorr is set.
Works only with filters which support this interface.

10.8.9   ConfigInterleaving as DirectShowConfigInterleavingMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the config interleaving object.
Notes: Lasterorr is set.
Works only with filters which support this interface.

10.8.10   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.8.11   EnumPins as DirectShowEnumPinsMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The EnumPins method enumerates the pins on this filter.
Example:

dim filter as DirectShowBaseFilterMBS // your device filter

dim pn as DirectShowEnumPinsMBS = srcfilter.EnumPins

// loop over them and show names of pins
dim p as DirectShowPinMBS = pn.NextObject
while p <>nil
dim name as string = p.Name

MsgBox name

p = pn.NextObject
wend

Notes: Lasterror is set.

10.8.12   FindPin(name as string) as DirectShowPinMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The FindPin method retrieves the pin with the specified identifier.
Notes: name: string that identifies the pin.

Returns the pin object.
Lasterror is set.

This method supports graph persistence. Use the DirectShowPinMBS.QueryId method to save a pin's state, and use this method to restore the state. The pin's identifier string is defined by the filter implementation. The identifier must be unique within the filter.

10.8.13   Info as DirectShowFilterInfoMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves information about the filter.
Notes: Lasterror is set.
Returns nil on error.

10.8.14   ShowPropertyDialog(parent as DesktopWindow, x as integer = 0, y as integer = 0, title as string = "")

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

Function: Shows properties dialog for the filter.
Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.8.15   ShowPropertyDialog(parent as window = nil, x as integer = 0, y as integer = 0, title as string = "")

Plugin Version: 19.1, Platform: Windows, Targets: Desktop only.

Function: Shows properties dialog for the filter.
Example:

dim srcfilter as DirectShowBaseFilterMBS // your filter

srcfilter.ShowPropertyDialog
MsgBox "Result: "+Str(srcfilter.Lasterror)+": "+srcfilter.LasterrorMessage

Notes: Parent: the parent window for the dialog.
X/Y: The offset of the dialog relative to the parent window.
Title: optional title for the dialog.

Sets lasterror property.
See also:

10.8.16   VendorInfo as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves a string containing vendor information.
Notes: Lasterror is set.
This method is optional; filters are not required to support it.

10.9 class DirectShowBindContextMBS

10.9.1 class DirectShowBindContextMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Provides access to a bind context, which is an object that stores information about a particular moniker binding operation.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr2

10.9.2 Methods

10.9.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to create a new bind context.

10.9.4 Properties

10.9.5 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IConfigAviMux interface.
(Read and Write property)

10.9.6 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.9.7   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.10   class DirectShowCaptureGraphBuilderMBS

10.10.1   class DirectShowCaptureGraphBuilderMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Builds capture graphs and other custom filter graphs.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd376359(v=vs.85).aspx
Blog Entries

- News from the MBS Xojo Plugins Version 21.1

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.1

- MBS Xojo Plugins, version 21.1pr1

- MBS Xojo Plugins, version 18.4pr4

- MBS Xojo / Real Studio Plugins, version 14.1pr3

Xojo Developer Magazine

- 19.3, page 10: News

10.10.2   Methods

10.10.3   AllocCapFile(FilePath as string, Size as UInt64)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The AllocCapFile method preallocates a capture file to a specified size. For best results, always capture to a defragmented, preallocated capture file that is larger than the size of the capture data.
Notes: FilePath: a string that contains the name of the file to create or resize.
Size: Size of the file to allocate, in bytes.

Lasterror is set.

This method fails if the file is read-only.
It is best to allocate as much space as possible—ideally, more than needed. However, this can result in a very large file that contains relatively little data. For example, a 1-gigabyte (GB) capture file might contain a few megabytes of captured video. Use the CopyCaptureFile method to copy the data into a new file. That method copies only the data and ignores the empty portion of the original file.
If you use this method to preallocate the file, call SetMode on the file-writer filter with the value zero. If the filter is set to AM_FILE_OVERWRITE, it will delete the preallocated file. Note that some file-writer filters do not support mode 0.

10.10.4   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

10.10.5   Crossbar(filter as DirectShowBaseFilterMBS) as DirectShowAMCross-barMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Finds a crossbar in the graph.
Notes: See also:
http://msdn.microsoft.com/en-us/library/windows/desktop/dd390991(v=vs.85).aspx

This function implements the FindInterface call with LOOK_UPSTREAM_ONLY and the filter you provide.

10.10.6   FindPin(Source as DirectShowBaseFilterMBS, PinDirection as Integer, Category as DirectShowGUIDMBS = nil, Type as DirectShowGUIDMBS = nil, Unconnected as boolean = false, Num as Integer = 0) as DirectShowPinMBS

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The FindPin method retrieves a particular pin on a filter, or determines whether a given pin matches the specified criteria.
Notes: Source: an interface on a filter, or to an interface on a pin.
PinDirection: Specifies the pin direction (input or output). See constants in DirectShowPinMBS class.
Category: A GUID that specifies one of the pin categories listed in Pin Property Set. To match any pin, regardless of category, set this parameter to nil.
Type: A major type GUID that specifies the media type. Use nil to match any media type.
Unconnected: Boolean value that specifies whether the pin must be unconnected. If true, the pin must be unconnected. If false, the pin can be connected or unconnected.
num: Zero-based index of the pin to retrieve, from the set of matching pins. If Source is a filter, and more than one pin matches the search criteria, this parameter specifies which pin to retrieve. If Source is a pin, this parameter is ignored.

Returns the pin object.
Lasterror is set.

If Source is a filter, the method searches for the nth pin on that filter that matches the search criteria, where

n is given by the num parameter. If the method finds a matching pin, it returns the pin.
If Source is a pin, the method tests that pin against the search criteria. If the pin matches the criteria, the method returns the pin. Otherwise, it returns nil.

Typically, an application will not need to use this method. It is provided for unusually complex tasks, when the RenderStream method cannot build the filter graph. Use this method to retrieve a desired pin from a capture filter, and then build the rest of the graph manually.
See also:

10.10.7   FindPin(Source as DirectShowPinMBS, PinDirection as Integer, Category as DirectShowGUIDMBS = nil, Type as DirectShowGUIDMBS = nil, Unconnected as boolean = false, Num as Integer = 0) as DirectShowPinMBS

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The FindPin method retrieves a particular pin on a filter, or determines whether a given pin matches the specified criteria.
Notes: Source: an interface on a filter, or to an interface on a pin.
PinDirection: Specifies the pin direction (input or output). See constants in DirectShowPinMBS class.
Category: A GUID that specifies one of the pin categories listed in Pin Property Set. To match any pin, regardless of category, set this parameter to nil.
Type: A major type GUID that specifies the media type. Use nil to match any media type.
Unconnected: Boolean value that specifies whether the pin must be unconnected. If true, the pin must be unconnected. If false, the pin can be connected or unconnected.
num: Zero-based index of the pin to retrieve, from the set of matching pins. If Source is a filter, and more than one pin matches the search criteria, this parameter specifies which pin to retrieve. If Source is a pin, this parameter is ignored.

Returns the pin object.
Lasterror is set.

If Source is a filter, the method searches for the nth pin on that filter that matches the search criteria, where n is given by the num parameter. If the method finds a matching pin, it returns the pin.
If Source is a pin, the method tests that pin against the search criteria. If the pin matches the criteria, the method returns the pin. Otherwise, it returns nil.

Typically, an application will not need to use this method. It is provided for unusually complex tasks, when the RenderStream method cannot build the filter graph. Use this method to retrieve a desired pin from a capture filter, and then build the rest of the graph manually.

See also:

- 10.10.6 FindPin(Source as DirectShowBaseFilterMBS, PinDirection as Integer, Category as Direct-ShowGUIDMBS = nil, Type as DirectShowGUIDMBS = nil, Unconnected as boolean = false, Num as Integer = 0) as DirectShowPinMBS   274

10.10.8   GetFiltergraph as DirectShowGraphBuilderMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the filter graph that the builder is using.
Notes: Sets lasterror.

10.10.9   GetStreamConfig(Category as DirectShowGUIDMBS, filter as Direct-ShowBaseFilterMBS) as DirectShowAMStreamConfigMBS

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the stream config object for this graph.
Example:

dim graph as DirectShowCaptureGraphBuilderMBS // your graph
dim filter as DirectShowBaseFilterMBS // your filter

dim StreamConfig as DirectShowAMStreamConfigMBS
StreamConfig = g.GetStreamConfig(DirectShowPinMBS.PIN_CATEGORY_STILL, filter)

Notes: Lasterror is set.
e.g. pass PIN_CATEGORY_CAPTURE or PIN_CATEGORY_PREVIEW for category, as defined in PinMBS class.
See also:

- 10.10.10 GetStreamConfig(preview as boolean, filter as DirectShowBaseFilterMBS) as DirectShowAMStream-ConfigMBS   276

10.10.10   GetStreamConfig(preview as boolean, filter as DirectShowBaseFilterMBS) as DirectShowAMStreamConfigMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the stream config object for this graph.
Notes: Lasterror is set.

Preview: Pass true to get the preview pin or pass false for the capture pin instead.
See also:

10.10.11 MEDIATYPE_Audio as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Audio.

10.10.12 MEDIATYPE_AUXLine21Data as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Line 21 data. Used by closed captions.

10.10.13 MEDIATYPE_Interleaved as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Interleaved audio and video. Used for Digital Video (DV).

10.10.14 MEDIATYPE_Midi as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: MIDI format.

10.10.15 MEDIATYPE_ScriptCommand as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Data is a script command, used by closed captions.

10.10.16   MEDIATYPE_Stream as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Byte stream with no time stamps.

10.10.17   MEDIATYPE_Text as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Text.

10.10.18   MEDIATYPE_Timecode as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Timecode data. Note: DirectShow does not provide any filters that support this media type.

10.10.19   MEDIATYPE_Video as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the media type GUIDs.
Notes: Video.

10.10.20   RenderStream(category as DirectShowGUIDMBS, Type as Direct-ShowGUIDMBS, Source as DirectShowBaseFilterMBS, Intermediate as DirectShowBaseFilterMBS = nil, Sink as DirectShowBaseFilterMBS = nil)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The RenderStream method connects an output pin on a source filter to a sink filter, optionally through an intermediate filter.
Notes: Lasterror is set:

Category: A GUID that specifies one of the pin categories listed in Pin Property Set. To match any pin, regardless of category, set this parameter to nil. Typical values include the following. PIN_CATE-

GORY_CAPTURE, PIN_CATEGORY_PREVIEW, PIN_CATEGORY_CC.
Type: A major-type GUID that specifies the media type of the output pin; or nil to use any pin, regardless of media type.
Source: Specifies a pointer to the starting filter for the connection, or to an output pin.
Intermediate: BaseFilter interface of an intermediate filter, such as a compression filter. Can be nil.
Sink: BaseFilter interface of a sink filter, such as a renderer or mux filter. If the value is nil, the method uses a default renderer (see Remarks).

see also:
http://msdn.microsoft.com/en-us/library/windows/desktop/dd311924(v=vs.85).aspx
See also:

- 10.10.21 RenderStream(category as DirectShowGUIDMBS, Type as DirectShowGUIDMBS, Source as DirectShowPinMBS, Intermediate as DirectShowBaseFilterMBS = nil, Sink as DirectShowBaseFilterMBS = nil) 279

10.10.21 RenderStream(category as DirectShowGUIDMBS, Type as DirectShowGUIDMBS, Source as DirectShowPinMBS, Intermediate as DirectShowBaseFilterMBS = nil, Sink as DirectShowBaseFilterMBS = nil)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The RenderStream method connects an output pin on a source filter to a sink filter, optionally through an intermediate filter.
Notes: Lasterror is set:

Category: A GUID that specifies one of the pin categories listed in Pin Property Set. To match any pin, regardless of category, set this parameter to nil. Typical values include the following. PIN_CATEGORY_CAPTURE, PIN_CATEGORY_PREVIEW, PIN_CATEGORY_CC.
Type: A major-type GUID that specifies the media type of the output pin; or nil to use any pin, regardless of media type.
Source: Specifies a pointer to the starting filter for the connection, or to an output pin.
Intermediate: BaseFilter interface of an intermediate filter, such as a compression filter. Can be nil.
Sink: BaseFilter interface of a sink filter, such as a renderer or mux filter. If the value is nil, the method uses a default renderer (see Remarks).

see also:
http://msdn.microsoft.com/en-us/library/windows/desktop/dd311924(v=vs.85).aspx
See also:

- 10.10.20 RenderStream(category as DirectShowGUIDMBS, Type as DirectShowGUIDMBS, Source as DirectShowBaseFilterMBS, Intermediate as DirectShowBaseFilterMBS = nil, Sink as DirectShowBaseFilterMBS = nil) 278

10.10.22   SetFiltergraph(graph as DirectShowGraphBuilderMBS)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Tells the graph builder object which filter graph to use.
Notes: Lasterror is set.

graph: specifies the filter graph to use for subsequent calls to the AddFilter method.

The graph builder will automatically create a filter graph if you don't call this method. If you call this method after the graph builder has created its own filter graph, the call will fail.

10.10.23   SetOutputFileName(Type as DirectShowGUIDMBS, FilePath as string)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates the rendering section of the filter graph, which will save bits to disk with the specified file name.
Notes: Lasterror is set.

Type: GUID representing the media subtype. Must be MEDIASUBTYPE_Avi.
FilePath: string containing the output file name.
filter: Optional, a filter representing the multiplexer filter.
Sink: Optional, a FileSinkFilter object representing the file writer.

This method inserts the multiplexer and the file writer into the filter graph and calls SetFileName to set the output file name.
See also:

- 10.10.24 SetOutputFileName(Type as DirectShowGUIDMBS, FilePath as string, byref filter as Direct-ShowBaseFilterMBS, byref sink as DirectShowFileSinkFilterMBS)   280

10.10.24   SetOutputFileName(Type as DirectShowGUIDMBS, FilePath as string, byref filter as DirectShowBaseFilterMBS, byref sink as DirectShow-FileSinkFilterMBS)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates the rendering section of the filter graph, which will save bits to disk with the specified file name.
Notes: Lasterror is set.

Type: GUID representing the media subtype. Must be MEDIASUBTYPE_Avi.
FilePath: string containing the output file name.
filter: Optional, a filter representing the multiplexer filter.
Sink: Optional, a FileSinkFilter object representing the file writer.

This method inserts the multiplexer and the file writer into the filter graph and calls SetFileName to set the output file name.
See also:

10.10.25   SetupHighestResolution(videoInputFilter as DirectShowBaseFilterMBS, preview as boolean = false)

Plugin Version: 18.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience function to query stream config and select the one with highest resolution.
Notes: Queries internally GetStreamConfig, walks over all VideoCaps, picks best and sets it as format.

10.10.26   Properties

10.10.27   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an ICaptureGraphBuilder interface.
(Read and Write property)

10.10.28   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.10.29   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.11 class DirectShowConfigAviMuxMBS

10.11.1 class DirectShowConfigAviMuxMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class configures the AVI Mux filter.
Notes: Applications can use this class to set the master stream and to create an AVI 1.0 index.

DirectShowConfigAviMuxMBS provides backward compatibility with older Video for Windows (VFW) Audio-Video Interleaved (AVI) index formats (idx1) as well as extended AVI 2.0 index formats (indx) to allow for file sizes greater than 1 gigabyte (GB). Set and retrieve the compatibility indexes by using the OutputCompatibilityIndex property.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.11.2 Methods

10.11.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.11.4 Properties

10.11.5 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IConfigAviMux interface.
(Read and Write property)

10.11.6 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.

(Read and Write property)

10.11.7   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.11.8   MasterStream as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the master stream that will be used to synchronize the other streams in the file.
Notes: Specifies the index of the stream, or –1 to indicate no master stream. The AVI Mux writes one stream for each connected input pin. Stream numbers are indexed from zero.
Lasterror is set.

If you are capturing audio and video from two different sources, use this method to synchronize the streams. Streams coming from separate capture sources may be captured at slightly different rates. If you specify a master stream, the AVI Mux adjusts the playback rates for the other streams, to compensate for any drift that might occur.
It is recommended to use the audio stream as the master stream, because minor adjustments to the video playback rate are less noticeable than changes to the audio playback rate. Also, modifying the audio playback rate will cause the audio to be resampled by the audio driver.
This method works by adjusting the dwScale and dwRate values in the AVISTREAMHEADER structure.
(Read and Write computed property)

10.11.9   OutputCompatibilityIndex as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set the AVI index format.
Notes: Lasterror is set.
True: Create an AVI 1.0 index, as well as an AVI 2.0 index.
False: Create an AVI 2.0 index, but not an AVI 1.0 index.

The AVI Mux filter always creates an AVI 2.0 index ('indx' format). If the value given in fOldIndex is TRUE, the AVI Mux also creates an AVI 1.0 index ('idx1' format), for backward compatibility with Video for Windows.
The AVI 2.0 index format allows for larger files, incremental growth of files, and minimized disk seeks.

(Read and Write computed property)

10.12   class DirectShowConfigInterleavingMBS

10.12.1   class DirectShowConfigInterleavingMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ConfigInterleaving interface controls how the AVI Mux filter interleaves audio and video samples.
Notes: Video-authoring applications that handle capturing should use this interface when they need to control how audio samples and video frames will be saved on a disk.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.12.2   Methods

10.12.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.12.4   Properties

10.12.5   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IConfigInterleaving interface.
(Read and Write property)

10.12.6   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.12.7 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.12.8 Mode as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The interleaving quality setting.
Notes: Lasterror is set.
See kInterleaving* constants.
(Read and Write computed property)

10.12.9 Constants

Interleaving Mode Constants

Constant	Value	Description
kInterleaveBuffered	3	Noninterleaved.
		This mode is equivalent to kInterleaveNone but uses less file space and system overhead.
kInterleaveCapture	1	Approximate interleaving with less overhead than kInterleaveFull.
		This mode is suitable for video capture. The AVI Mux attempts to use unbuffered, overlapped write operations. Unless the interleaving parameters are configured properly, however, frames may be dropped if one stream blocks while it waits for data from another stream. In particular, audio buffers should be less than .5 second, or else the video stream will block for excessive periods of time.
kInterleaveFull	2	Full, precise interleaving of audio samples and video frames.
		Streams will block indefinitely, waiting for equal amounts of data before interleaving. This mode is suitable for authoring and playback.
kInterleaveNone	0	Noninterleaved.
		Frames are written in the order they arrive. Files must be interleaved for playback at a later time. In this mode, the AVI Mux filter attempts to use unbuffered, overlapped write operations, to increase throughput.

10.13   class DirectShowDVInfoMBS

10.13.1   class DirectShowDVInfoMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes the format of a digital video (DV) stream.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.4pr7

10.13.2   Methods

10.13.3   Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.13.4   Properties

10.13.5   DVAAuxCtl as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the AAUX source control Pack for the first audio block.
Notes: (Read and Write property)

10.13.6   DVAAuxCtl1 as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the AAUX source control pack for the second audio block.
Notes: (Read and Write property)

10.13.7   DVAAuxSrc as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the audio auxiliary (AAUX) source pack for the first audio block.
Notes: (Read and Write property)

10.13.8   DVAAuxSrc1 as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the AAUX source pack for the second audio block.
Notes: (Read and Write property)

10.13.9   DVVAuxCtl as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the VAUX source control pack.
Notes: (Read and Write property)

10.13.10   DVVAuxSrc as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the video auxiliary (VAUX) source pack.
Notes: (Read and Write property)

10.14　class DirectShowEnumMonikerMBS

10.14.1　class DirectShowEnumMonikerMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enumerates the monikers in a table of monikers.
Blog Entries

- MBS Xojo Plugins, version 21.5pr2

10.14.2　Methods

10.14.3　Clone as DirectShowEnumMonikerMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new enumerator that contains the same enumeration state as the current one.
Notes: This method makes it possible to record a particular point in the enumeration sequence and then return to that point at a later time. The caller must release this new enumerator separately from the first enumerator.

Returns nil on any error.
Lasterror is set.

10.14.4　CLSID_AudioCompressorCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Audio Compressors

10.14.5　CLSID_AudioInputDeviceCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Audio Capture Sources

10.14.6 CLSID_AudioRendererCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Audio Renderers

10.14.7 CLSID_DeviceControlCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Device Control Filters

10.14.8 CLSID_DVDHWDecodersCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: WDM Stream Decompression Devices
This category contains hardware DVD decoders.

10.14.9 CLSID_LegacyAmFilterCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: DirectShow Filters

10.14.10 CLSID_MidiRendererCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Midi Renderers

10.14.11 CLSID_TransmitCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: External Renderers

10.14.12   CLSID_VideoCompressorCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Video Compressors

10.14.13   CLSID_VideoInputDeviceCategory as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow filter categories.
Notes: Video Capture Sources

10.14.14   Constructor

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to find filters.
Notes: Uses IFilterMapper2::EnumMatchingFilters to find installed filters in the registry.
Lasterror is set on failure.
See also:

10.14.15   Constructor(clsidDeviceClass as DirectShowGUIDMBS)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an enumerator for a specified device category.
Notes: This method makes it possible to record a particular point in the enumeration sequence and then return to that point at a later time. The caller must release this new enumerator separately from the first enumerator.

Lasterror is set.
See also:

10.14.16   Destructor

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.

10.14.17   NextObject as DirectShowMonikerMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the next item in the enumeration sequence.
Notes: Returns nil on any error.
Lasterror is set.

10.14.18   Reset

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumeration sequence to the beginning.
Notes: Lasterror is set.
There is no guarantee that the same set of objects will be enumerated after the reset operation has completed. A static collection is reset to the beginning, but it can be too expensive for some collections, such as files in a directory, to guarantee this condition.

10.14.19   Skip(n as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips over the specified number of items in the enumeration sequence.
Notes: Lasterror is set.

10.14.20   Properties

10.14.21   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IEnumMoniker interface.
(Read and Write property)

10.14.22   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.14.23   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.15 class DirectShowEnumPinsMBS

10.15.1 class DirectShowEnumPinsMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enumerates pins on a filter.
Notes: The BaseFilterMBS.EnumPins method returns this interface. It is based on the standard Component Object Model (COM) enumerators.
The filter graph manager uses this interface when it connects filters. Applications can use it to retrieve pins on a filter. For more information, see Enumerating Objects in a Filter Graph.
If the number of pins on the filter changes, some methods on this interface return VFW_E_ENUM_OUT_OF_SYNC. Call the Reset method to resynchronize the enumerator.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.15.2 Methods

10.15.3 Clone as DirectShowEnumPinsMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Clone method makes a copy of the enumerator with the same enumeration state.
Notes: Lasterror is set. Returns nil on any error.

If the number of pins changes, the enumerator is no longer consistent with the filter, and the method returns VFW_E_ENUM_OUT_OF_SYNC. Discard any data obtained from previous calls to the enumerator, because it might be invalid. Update the enumerator by calling the IEnumPins::Reset method. You can then call the Clone method safely.

10.15.4 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.15.5 NextObject as DirectShowPinMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Next method retrieves the next pin in the enumeration sequence.
Notes: Lasterror is set. Returns nil on any error.

If the number of pins changes, the enumerator is no longer consistent with the filter, and the method returns VFW_E_ENUM_OUT_OF_SYNC. Discard any data obtained from previous calls to the enumerator, because it might be invalid. Update the enumerator by calling the IEnumPins::Reset method. You can then call the Next method safely.

10.15.6   Reset

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Reset method resets the enumeration sequence to the beginning.
Notes: Lasterror is set.

10.15.7   Skip(n as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Skip method skips over a specified number of pins.
Notes: n: Number of pins to skip.

Lasterror is set.

If the number of pins changes, the enumerator is no longer consistent with the filter, and the method returns VFW_E_ENUM_OUT_OF_SYNC. Discard any data obtained from previous calls to the enumerator, because it might be invalid. Update the enumerator by calling the IEnumPins::Reset method. You can then call the Skip method safely.

10.15.8   Properties

10.15.9   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IEnumPins interface.
(Read and Write property)

10.15.10   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.15.11 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.16   class DirectShowFileSinkFilterMBS

10.16.1   class DirectShowFileSinkFilterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The FileSinkFilter interface is implemented on filters that write media streams to a file.
Notes: A file sink filter in a video capture filter graph, for instance, writes the output of the video compression filter to a file. Typically, the application running this filter graph should enable the user to enter the name of the file to be written to. This interface enables the communication of this information.
If a filter needs the name of an output file, it should expose this interface to allow an application to set the file name. Note that there is currently no base class implementation of this interface.
Any application that must set the name of the file into which the file sink filter will write should use this interface to get and set the file name.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.16.2   Methods

10.16.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.16.4   MEDIASUBTYPE_Asf as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The media sub type for ASF video files.

10.16.5   MEDIASUBTYPE_Avi as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The media sub type for AVI video files.

10.16.6 Properties

10.16.7 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IFileSinkFilter interface.
(Read and Write property)

10.16.8 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.16.9 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.17 class DirectShowFilterGraphMBS

10.17.1 class DirectShowFilterGraphMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class provides methods for building a filter graph.
Notes: An application can use it to add filters to the graph, connect or disconnect filters, remove filters, and perform other basic operations. However, the DirectShowGraphBuilderMBS interface inherits from this interface and provides additional methods that are more sophisticated. Therefore, applications should use DirectShowGraphBuilderMBS rather than using DirectShowFilterGraphMBS directly.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.17.2 Methods

10.17.3 AddFilter(SourceFilter as DirectShowBaseFilterMBS, Name as string = "")

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a filter to the graph.
Notes: Filter: The filter to add.
Name: name for filter.

Lasterror is set.

The name of the filter can be "", in which case the Filter Graph Manager generates a name. If the name is not "" and is not unique, this method will modify the name in an attempt to generate a new unique name. If this is successful, this method sets lasterror to VFW_S_DUPLICATE_NAME. If it cannot generate a unique name, it sets lasterror to VFW_E_DUPLICATE_NAME.
AddFilter calls the filter's JoinFilterGraph method to inform the filter that it has been added. AddFilter must be called before attempting to use the Connect, ConnectDirect, or Render method to connect or render pins belonging to the added filter.
The Filter Graph Manager holds a reference count on the filter until the filter is removed from the graph or the Filter Graph Manager is released.

10.17.4 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.17.5 SetDefaultSyncSource

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the reference clock to the default clock.
Notes: Lasterror is set.

This method instructs the Filter Graph Manager to choose a reference clock using its default algorithm. For more information about the algorithm that it uses, see Reference Clocks.
Usually you do not need to call this method, because the Filter Graph Manager automatically selects a clock. However, if you call SetSyncSource to override the clock, you can use SetDefaultSyncSource to restore the default clock.
This method fails if the filter graph is running or paused.

10.17.6 Properties

10.17.7 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IFilterGraph interface.
(Read and Write property)

10.17.8 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.17.9 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.18   class DirectShowFilterInfoMBS

10.18.1   class DirectShowFilterInfoMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for information about a filter.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.18.2   Methods

10.18.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.18.4   Properties

10.18.5   Graph as DirectShowFilterGraphMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns filter graph.
Notes: If the filter is member of a filter graph, this is a reference to the filter graph's FilterGraph interface.
If the filter is not a member of a filter graph, this value of this member is nil.
(Read and Write property)

10.18.6   Name as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the filter.
Notes: (Read and Write property)

10.19 class DirectShowGraphBuilderMBS

10.19.1 class DirectShowGraphBuilderMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class provides methods that enable an application to build a filter graph.
Notes: The GraphBuilder interface inherits from the FilterGraph interface. FilterGraph provides basic operations, such as adding a filter to the graph or connecting two pins. GraphBuilder adds further methods that construct graphs from partial information. For example, the RenderFile method builds a graph for file playback, given the name of the file. The Render method renders data from an output pin by connecting new filters to the pin.

Using these methods, an application does not need to specify every filter and pin connection in the graph. Instead, the Filter Graph Manager selects filters that are registered on the user's system, adds them to the graph, and connects them.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd390085(v=vs.85).aspx
Subclass of the DirectShowFilterGraphMBS class.
Blog Entries

- MBS Real Studio Plugins, version 12.3pr8

10.19.2 Methods

10.19.3 Abort

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Abort method requests the Filter Graph Manager to halt its current task as quickly as possible.
Notes: The current task may or may not fail to complete. Possibly the fastest option for the Filter Graph Manager is to complete the task.

Lasterror is set.

10.19.4 AddSourceFilter(FileName as string, FilterName as string) as DirectShowBaseFilterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a source filter for a specified file to the filter graph.

Notes: FileName: the name of the file to load.
FilterName: name for the source filter.

Lasterror is set.

This method searches for an installed filter that can read the specified file. If it finds one, the method adds it to the filter graph and returns a pointer to the filter's IBaseFilter interface. To determine the media type and compression scheme of the file, the Filter Graph Manager reads the first few bytes of the file, looking for specific patterns of bytes, as documented in the article Registering a Custom File Type.

The application is responsible for building the rest of the filter graph. To do so, call EnumPins to enumerate the output pins on the source filter. Then use either the Connect method or the Render method.

If the method succeeds, the BaseFilter interface has an outstanding reference count. The caller must release the interface.

To render a file for default playback, use the RenderFile method.

The Filter Graph Manager holds a reference count on the filter until the filter is removed from the graph or the Filter Graph Manager is released.

10.19.5   Connect(pinOut as DirectShowPinMBS, pinIn as DirectShowPinMBS)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Connects the two pins, using intermediates if necessary.
Notes: pinOut: the output pin.
pinIn: the input pin.

Lasterror is set.

This method connects two pins directly or indirectly, adding intermediate filters if necessary. The method starts by attempting a direct connection. If that fails, it tries to use any filters that are already in the filter graph and have unconnected input pins. (It enumerates these in an arbitrary order.) If that fails, it searches for filters in the registry, and tries them in order of merit.

During the connection process, the Filter Graph Manager ignores pins on intermediate filters if the pin name begins with a tilde (
textasciitilde).

see also

http://msdn.microsoft.com/en-us/library/windows/desktop/dd390088(v=vs.85).aspx

10.19.6 ConnectFilters(pinOut as DirectShowPinMBS, dest as DirectShowBase-FilterMBS)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience function to connect an output pin to a filter.
Notes: Lasterror is set.
See also:

10.19.7 ConnectFilters(source as DirectShowBaseFilterMBS, dest as Direct-ShowBaseFilterMBS)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience function to connect two filters.
Notes: Lasterror is set.
See also:

10.19.8 ConnectFilters(source as DirectShowBaseFilterMBS, pinIn as Direct-ShowPinMBS)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience function to connect a filter to an input pin.
Notes: Lasterror is set.
See also:

10.19.9 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

10.19.10 MediaControl as DirectShowMediaControlMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries related media control object.
Notes: Lasterror is set.

10.19.11 MediaEventEx as DirectShowMediaEventExMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries related media event object.
Notes: Lasterror is set.

10.19.12 Render(pinOut as DirectShowPinMBS)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Render method builds a filter graph that renders the data from a specified output pin.
Notes: pinOut: an output pin.
Lasterror is set.

This method renders the data from a specified output pin, adding new filters to the graph as needed. Filters are tried in the same order as for the Connect method. For more information, see Intelligent Connect. During the connection process, the Filter Graph Manager ignores pins on intermediate filters if the pin name begins with a tilde (
textasciitilde).

10.19.13 RenderFile(FilePath as string)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Builds a filter graph that renders the specified file.
Notes: FilePath: string that contains the name of a media file.

Lasterror is set.

If the FilePath parameter specifies a media file, the method builds a filter graph for default playback. First it adds a source filter that can read the file, using the same process as the AddSourceFilter method. Then it renders the output pins on the source filter, adding intermediate filters if necessary. It tries filters in the same order as the Connect method.

During the connection process, the Filter Graph Manager ignores pins on intermediate filters if the pin name begins with a tilde (

textasciitilde).

Note that the RenderFile method does not remove any filters from the graph. If you call RenderFile twice, the second call simply adds more filters to the graph. When you run the graph, both sources will play at the same time.

10.19.14   SetLogFile(FilePath as string)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetLogFile method sets the file for logging actions taken when attempting to perform an operation.

Notes: This method is for use in debugging; it is intended to help you determine the cause of any failure to automatically build a filter graph.

Lasterror is set.

10.19.15   VideoWindow as DirectShowVideoWindowMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the video window related tot this graph builder.

Notes: Lasterror is set.

10.20 class DirectShowGUIDMBS

10.20.1 class DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a Windows unique ID.
Notes: If you need to validate a GUID or UUID, please check the IsGUID function in our FAQ.
Blog Entries

- News from the MBS Xojo Plugins Version 21.1

- MBS Xojo Plugins, version 21.1pr1

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.0

- MBS Xojo Plugins, version 18.0pr6

10.20.2 Methods

10.20.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a GUID with only zeros.
Example:

dim g as new DirectShowGUIDMBS

MsgBox g.DisplayString

See also:

10.20.4 Constructor(Value as String)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates GUID based on a text string.
Notes: Raises exception if text has wrong format.
See also:

10.20.5 Constructor(value1 as Integer, value2 as Integer, value3 as Integer, value4 as Integer, value5 as Integer, value6 as Integer, value7 as Integer, value8 as Integer, value9 as Integer, value10 as Integer, value11 as Integer, value12 as Integer, value13 as Integer, value14 as Integer, value15 as Integer, value16 as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new GUID with the given byte values.
Example:

dim g as new DirectShowGUIDMBS(&h14, &h3e, &h4e, &h83, &h64, &h97, &h11, &hd2, &ha2, &h31, &h00, &hc0, &h4f, &ha3, &h18, &h09)

MsgBox g.DisplayString

See also:

- 10.20.3 Constructor 308

- 10.20.4 Constructor(Value as String) 308

10.20.6 Equal(other as DirectShowGUIDMBS) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether two GUIDs are equal.
Notes: Returns true if both items are equals.

10.20.7 Operator_Convert as String

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Converts GUID to text.
Notes: Same as DisplayName.
See also:

10.20.8   Operator_Convert(text as String)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Converts text to GUID.
Notes: Same as Constructor.
Raises exception if text has wrong format.
See also:

10.20.9   Parse(GUID as String) as DirectShowGUIDMBS

Plugin Version: 18.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Parses GUID string into a GUID object.

10.20.10   Properties

10.20.11   Data as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The UID as binary string.
Notes: (Read and Write property)

10.20.12   DisplayString as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID well formatted.
Notes: (Read only property)

10.20.13   Memory as MemoryBlock

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set GUID with memory block.
Notes: (Read and Write property)

10.20.14 Ptr as Ptr

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pointer to raw GUID data.
Notes: (Read only property)

10.20.15 Byte(index as Integer) as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Read or write the byte value.
Example:

dim g as new DirectShowGUIDMBS

g.Byte(1) = 65

MsgBox str(g.Byte(1)) // shows 65

Notes: (Read and Write computed property)

10.21   class DirectShowMediaControlMBS

10.21.1   class DirectShowMediaControlMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Provides methods for controlling the flow of data through the filter graph.
Notes: It includes methods for running, pausing, and stopping the graph. The Filter Graph Manager implements this interface.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.21.2   Methods

10.21.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.21.4   GetState(msTimeout as Integer = -1) as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the state of the filter graph—paused, running, or stopped.
Notes: msTimeout: Duration of the time-out, in milliseconds, or INFINITE (-1) to specify an infinite time-out.

State transitions are not necessarily synchronous. Therefore, when you call this method, the filter graph might be in transition to a new state. In that case, the method blocks until the transition completes or until the specified time-out elapses.

Lasterror is set.

Applications can use this method to determine whether playback has started after a call to Run. Generally, applications should have their own mechanism for tracking which state they have put the filter graph into. Applications typically use the current state to determine which user interface controls are enabled or disabled. For example, once the graph goes into the running state, the application might disable a "Play" button and enable "Stop" and "Pause" buttons.
If the filter graph is in a transition to a new state, the returned state is the new state, not the previous state. This method returns an error if there is a call on another thread to change the state while this method is blocked.
Avoid specifying a time-out of INFINITE, because threads cannot process messages while waiting in Get-

State. If you call GetState from the thread that processes Windows messages, specify small wait times on the call in order to remain responsive to user input. This is especially important when the source is streaming over a network or from the Internet because state transitions in these environments can take significantly more time to complete.

10.21.5 Pause

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pauses all the filters in the filter graph.
Notes: Lasterror is set.

Pausing the filter graph cues the graph for immediate rendering when the graph is next run. While the graph is paused, filters process data but do not render it. Data is pushed through the graph and processed by transform filters as far as buffering permits, but renderer filters do not render the data. However, video renderers display a static poster frame of the first sample.

If the method returns S_FALSE, call the GetState method to wait for the state transition to complete, or to check if the transition has completed. When you call Pause to display the first frame of a video file, always follow it immediately with a call to GetState to ensure that the state transition has completed. Failure to do this can result in the video rectangle being painted black.
If the method fails, it stops the graph before returning.

10.21.6 RenderFile(FilePath as string)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Builds a filter graph that renders the specified file.
Notes: Lasterror is set.

filePath: Specifies the name of the file to load.

10.21.7 Run

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs all the filters in the filter graph.
Notes: While the graph is running, data moves through the graph and is rendered.
Lasterror is set.

If the filter graph is stopped, this method pauses the graph before running. If the graph is already running, the method returns S_OK but has no effect.

The graph runs until the application calls the Pause or Stop method. When playback reaches the end of the stream, the graph continues to run, but the filters do not stream any more data. At that point, the application can pause or stop the graph. For information about the end-of-stream event, see Pause and EC_COMPLETE.

This method does not seek to the beginning of the stream. Therefore, if you run the graph, pause it, and then run it again, playback resumes from the paused position. If you run the graph after it has reached the end of the stream, nothing is rendered. To seek the graph, use the MediaSeeking interface.

If method returns S_FALSE, it means that the method returned before all of the filters switched to a running state. The filters will complete the transition after the method returns. Optionally, you can wait for the transition to complete by calling the GetState method with a timeout value. However, this is not required. If the Run method returns an error code, it means that one or more filters failed to run. However, some filters might be in a running state. In a multistream graph, entire streams might be playing successfully. Typically the application would tear down the graph and report an error in this case.

10.21.8   Stop

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Stops all the filters in the graph.
Notes: Lasterror is set.

If the graph is running, this method pauses the graph before stopping it. While paused, video renderers can copy the current frame to display as a poster frame.

This method does not seek to the beginning of the stream. If you call this method and then call the Run method, playback resumes from the stopped position. To seek, use the IMediaSeeking interface.

The Filter Graph Manager pauses all the filters in the graph, and then calls the Stop method on all filters, without waiting for the pause operations to complete. Therefore, some filters might have their Stop method called before they complete their pause operation. If you develop a custom rendering filter, you might need to handle this case by pausing the filter if it receives a stop command while in a running state. However, most filters do not need to take any special action in this regard.

10.21.9   StopWhenReady

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pauses the filter graph, allowing filters to queue data, and then stops the filter graph.
Notes: Lasterror is set.

This method is useful if you want to seek the filter graph while the graph is stopped. As long as the filter graph is stopped, changes in the current position do not repaint the video window with a new frame. Therefore, calling SetPositions does not update the video window. To update the window after the seek operation, call StopWhenReady. This method transitions the graph to a paused state, waits for the pause operation to complete, and then transitions the graph back to stopped. The pause operation queues data in the graph, so that the video renderer receives and displays the new frame.

This method is asynchronous. It waits on a separate thread for the pause to complete. The calling thread does not block, which enables the application to respond to user input. When the method returns, the logical state of the graph is stopped, even before the pause operation completes. If you call the GetState method at this point, it returns State_Stopped.

If the application issues another state-change command (such as pause, run, or seek) before the pause operation completes, the new command cancels the pending stop command. The pause operation completes, but the graph does not stop.

10.21.10   Properties

10.21.11   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IMediaControl interface.
(Read and Write property)

10.21.12   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.21.13   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.21.14   Constants

State Constants

Constant	Value	Description
kStatePaused	1	Paused
kStateRunning	2	Running
kStateStopped	0	Stopped.

10.22   class DirectShowMediaEventExMBS

10.22.1   class DirectShowMediaEventExMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for extended options for media event handling.
Notes: The DirectShowMediaEventExMBS interface inherits the DirectShowMediaEventMBS interface, which contains methods for retrieving event notifications and for overriding the filter graph's default handling of events. MediaEventEx adds methods that enable an application window to receive messages when events occur.
Subclass of the DirectShowMediaEventMBS class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.22.2   Methods

10.22.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.22.4   Properties

10.22.5   NotifyFlags as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enables or disables event notifications.
Notes: Value indicating whether to enable or disable event notifications. Must be one of the following values: 0 = Enable event notifications or AM_MEDIAEVENT_NONOTIFY = Disable event notifications.

Lasterror is set.

By default, the Filter Graph Manager posts event notifications for the application. If the NotifyFlags parameter is AM_MEDIAEVENT_NONOTIFY, the Filter Graph Manager clears any pending event notifications from the queue, and does not post any new ones.
If event notifications are disabled, the handle returned by the GetEventHandle method is signaled at the end of each stream—that is, whenever the Filter Graph Manager receives an EC_COMPLETE event.
(Read and Write computed property)

10.22.6   Constants

Constants

Constant	Value	Description
AM_MEDIAEVENT_NONOTIFY	1	One of the notify flags constant. Pass this value to NotifyFlags to disable event notification.

10.23 class DirectShowMediaEventMBS

10.23.1 class DirectShowMediaEventMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for media event handling.
Notes: The MediaEvent interface contains methods for retrieving event notifications and for overriding the Filter Graph Manager's default handling of events. The MediaEventEx interface inherits this interface and extends it.

The Filter Graph Manager implements this interface. Applications can use it to respond to events that occur in the filter graph, such as the end of a stream or a rendering error. Filters post events to the filter graph using the IMediaEventSink interface.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.23.2 Methods

10.23.3 CancelDefaultHandling(eventCode as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Cancels the Filter Graph Manager's default handling for a specified event.
Notes: The event notification is passed to the application.
Lasterror is set.

To restore the default handling for an event, call the RestoreDefaultHandling method with the event code.

10.23.4 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.23.5 FreeEventParams(eventCode as Integer, Param1 as Integer, Param2 as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Frees resources associated with the parameters of an event.
Notes: EventCode: Event code.

Param1: First event parameter.
Param2: Second event parameter.

Lasterorr is set.
After you call the GetEvent method to retrieve an event notification, you must call FreeEventParams. This method frees any resources that were allocated for the event parameters. Pass in the same variables used for the GetEvent call.

10.23.6   RestoreDefaultHandling(eventCode as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Restores the Filter Graph Manager's default handling for a specified event.
Notes: Lasterror is set.
By default, the Filter Graph Manager handles some events (such as EC_REPAINT) without passing them to the application. If you call the CancelDefaultHandling method to override the default handling for an event, you can restore the default behavior by calling RestoreDefaultHandling with the same event code.

10.23.7   Properties

10.23.8   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IMediaEvent interface.
(Read and Write property)

10.23.9   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.23.10   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.23.11 Constants

Constants

Constant	Value	Description
AM_MEDIAEVENT_NONOTIFY	1	One of the notify flags constant.
		Pass this value to NotifyFlags to disable event notification.

10.24   class DirectShowMediaFilterMBS

10.24.1   class DirectShowMediaFilterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The MediaFilter interface controls the streaming state of a filter.
Notes: All DirectShow filters implement this interface. It provides methods for switching the filter between states (stopped, paused, and running); for retrieving the filter's current state; and for setting a reference clock. Applications should not call MediaFilter methods on filters.

The Filter Graph Manager also exposes this interface. Applications can use the SetSyncSource method to set the graph reference clock, and GetSyncSource to retrieve the clock. Applications should not call the other methods on this interface. Instead, use the corresponding methods on the MediaControl interface.
The BaseFilter interface inherits from MediaFilter.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

10.24.2   Methods

10.24.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.24.4   Pause

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Pause method pauses the filter.
Notes: Lasterror is set.

When a filter is paused, it can receive, process, and deliver samples. However, a renderer filter will only accept one sample while paused. Therefore, when the filter graph is paused, samples move through the graph until the first sample reaches the renderer. At that point, streaming is paused until the Run method is called. Video renderers display the first sample as a still frame.

Live capture filters do not deliver any samples while paused, only while running.

The state transition might be asynchronous. If the method returns before the transition completes, the lasterror value is 1. A renderer filter does not complete the transition to paused until either (1) it receives one sample, or (2) it receives an end-of-stream notification. While the state transition is pending, State gives

lasterror VFW_S_STATE_INTERMEDIATE.

10.24.5   Run(StartTime as Int64)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Run method runs the filter.
Notes: StartTime: Reference time corresponding to stream time 0.
Lasterror is set.

When a filter is running, it can receive, process, and deliver samples. Source filters generate new samples, and renderer filters render them.
The state transition might be asynchronous. If the method returns before the transition completes, the lasterror value is 1.
Stream time is calculated as the current reference time minus StartTime. To calculate when a media sample should be rendered, the renderer compares the time stamp with the current stream time. Thus, a media sample with a time stamp of zero should be rendered at time StartTime. For more information, see Time and Clocks in DirectShow.
When an application calls the Run method, the Filter Graph Manager calls Run on each filter. It sets the value of StartTime slightly in the future, to account for graph latency.

10.24.6   Stop

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Stop method stops the filter.
Notes: Lasterror is set.
When a filter is stopped, it does not process or deliver any samples, and it rejects samples from upstream filters.
The state transition might be asynchronous. If the method returns before the transition completes, the lasterror value is 1.
This method always sets the filter's state to kStateStopped, even if the method returns an error code.

10.24.7   Properties

10.24.8   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IMediaFilter interface.

(Read and Write property)

10.24.9   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.24.10   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.24.11   Constants

Filter State Constants

Constant	Value	Description
kStatePaused	1	Paused. The filter is processing data, but not rendering it.
kStateRunning	2	Running. The filter is processing and rendering data.
kStateStopped	0	Stopped. The filter is not processing data.

10.25 class DirectShowMediaTypeMBS

10.25.1 class DirectShowMediaTypeMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes the format of a media sample.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd373477(v=vs.85).aspx
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 18.4pr4

- MBS Xojo / Real Studio Plugins, version 13.4pr7

10.25.2 Methods

10.25.3 Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.25.4 SetHeight(value as integer) as boolean

Plugin Version: 18.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets height of video info header.
Notes: Convenience function which works with VideoInfoHeader and VideoInfoHeader2 to set the value.

10.25.5 SetWidth(value as integer) as boolean

Plugin Version: 18.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets width of video info header.
Notes: Convenience function which works with VideoInfoHeader and VideoInfoHeader2 to set the value.

10.25.6   Properties

10.25.7   DVINFO as DirectShowDVInfoMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries format data as DVInfo.
Notes: Returns nil if format is not a DVInfo.
(Read only property)

10.25.8   FixedSizeSamples as Boolean

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: If true, samples are of a fixed size.
Notes: This field is informational only. For audio, it is generally set to true. For video, it is usually true for uncompressed video and false for compressed video.
(Read and Write property)

10.25.9   FormatType as DirectShowGUIDMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: GUID that specifies the structure used for the format block.
Notes: (Read and Write property)

10.25.10   Handle as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

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

10.25.11   Height as Integer

Plugin Version: 18.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries height of video info header.
Notes: Convenience function which works with VideoInfoHeader and VideoInfoHeader2 to query the value.
(Read only property)

10.25.12 MajorType as DirectShowGUIDMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Globally unique identifier (GUID) that specifies the major type of the media sample. F
Notes: or a list of possible major types, see Media Types.
http://msdn.microsoft.com/en-us/library/windows/desktop/dd390670(v=vs.85).aspx
(Read and Write property)

10.25.13 SampleSize as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Size of the sample in bytes. For compressed data, the value can be zero.
Notes: (Read and Write property)

10.25.14 SubType as DirectShowGUIDMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: GUID that specifies the subtype of the media sample.
Notes: For a list of possible subtypes, see Media Types. For some formats, the value might be MEDIA-SUBTYPE_None, which means the format does not require a subtype.
http://msdn.microsoft.com/en-us/library/windows/desktop/dd390670(v=vs.85).aspx
(Read and Write property)

10.25.15 TemporalCompression as Boolean

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: If true, samples are compressed using temporal (interframe) compression.
Notes: A value of true indicates that not all frames are key frames. This field is informational only.
(Read and Write property)

10.25.16 VideoInfoHeader as DirectShowVideoInfoHeaderMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries format data as VideoInfoHeader.
Notes: Returns nil if format is not a VideoInfoHeader.
(Read only property)

10.25.17   VideoInfoHeader2 as DirectShowVideoInfoHeader2MBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries format data as VideoInfoHeader2.
Notes: Returns nil if format is not a VideoInfoHeader2.
(Read only property)

10.25.18   WaveFormat as DirectShowWaveFormatMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries format data as WaveFormat.
Notes: Returns nil if format is not a WaveFormat.
(Read only property)

10.25.19   Width as Integer

Plugin Version: 18.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries width of video info header.
Notes: Convenience function which works with VideoInfoHeader and VideoInfoHeader2 to query the value.
(Read only property)

10.26 class DirectShowMonikerMBS

10.26.1 class DirectShowMonikerMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The moniker class to query information about devices.
Notes: In this plugin we use monikers to enumerate filters and devices in order to connect with BindBaseFilter method.

Enables you to use a moniker object, which contains information that uniquely identifies a COM object. An object that has a pointer to the moniker object's IMoniker interface can locate, activate, and get access to the identified object without having any other specific information on where the object is actually located in a distributed system.

Monikers are used as the basis for linking in COM. A linked object contains a moniker that identifies its source. When the user activates the linked object to edit it, the moniker is bound; this loads the link source into memory.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/ms679705(v=vs.85).aspx
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 12.1pr2](#)

10.26.2 Methods

10.26.3 BindBaseFilter as DirectShowBaseFilterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Binds moniker to a base filter.
Notes: Lasterror is set.

This implements the primary function of a moniker, which is to locate the object identified by the moniker and return a pointer to one of its interfaces.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/ms691433(v=vs.85).aspx

10.26.4   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.26.5   DisplayName(BindContext as DirectShowBindContextMBS = nil) as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the display name for the moniker.
Notes: BindContext: The DirectShowBindContextMBS interface on the bind context to be used in this operation. The bind context caches objects bound during the binding process, contains parameters that apply to all operations using the bind context, and provides the means by which the moniker implementation should retrieve information about its environment. If you provide nil, the plugin creates a temporary context.

GetDisplayName provides a string that is a displayable representation of the moniker. A display name is not a complete representation of a moniker's internal state; it is simply a form that can be read by users. As a result, it is possible (though rare) for two different monikers to have the same display name.

This is not a text to show to user.

10.26.6   EnumMonikers(forward as boolean) as DirectShowEnumMonikerMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves enumerator for the components of a composite moniker.
Notes: Lasterror is set.

10.26.7   Hash as UInt32

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a hash value using the internal state of the moniker.
Notes: Lasterror is set.

10.26.8   IsEqual(other as DirectShowMonikerMBS) as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Determines whether this moniker is identical to the specified moniker.
Notes: Returns true if both are identical.
Lasterror is set.

10.26.9 Properties(BindContext as DirectShowBindContextMBS = nil) as DirectShowPropertyBagMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates property bag with properties of the moniker.
Notes: Lasterror is set.

10.26.10 Properties

10.26.11 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IMoniker interface.
(Read and Write property)

10.26.12 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.26.13 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.27   class DirectShowNullRendererMBS

10.27.1   class DirectShowNullRendererMBS

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The null renderer filter.
Notes: The Null Renderer filter is a renderer that discards every sample it receives, without displaying or rendering the sample data.

Use this filter when an output pin in the graph requires a downstream connection, but you do not wish to render the data from that pin. By connecting the output pin to the Null Renderer, you complete the connection without rendering the data.
Subclass of the DirectShowBaseFilterMBS class.
Blog Entries

- MBS Real Studio Plugins, version 12.3pr8

10.27.2   Methods

10.27.3   Constructor

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new Null filter.
Notes: On success the handle property is not zero.

10.28 class DirectShowPinMBS

10.28.1 class DirectShowPinMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This interface is exposed by all input and output pins.
Notes: The filter graph manager uses this interface to connect pins and perform flushing operations. Applications can use this interface to query the pin for information.
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 21.1

- MBS Xojo Plugins Version 21.0 News

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0

- MBS Xojo Plugins, version 21.0pr8

Xojo Developer Magazine

- 21.1, page 35: News from MBS Xojo Plugins, What's up with MonkeyBread Software by Stefanie Juchmes

10.28.2 Methods

10.28.3 Accept(Type as DirectShowMediaTypeMBS) as Boolean

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Accept method determines whether the pin accepts a specified media type.
Notes: Lasterror is set.

A return value of true indicates that the pin will accept the media type, either on the next sample, or after a pin reconnection. The implementation should take into account the current state of the filter, including connections on other pins, and any properties that can be set on the filter.

Any other return value, including false, means that the pin rejects the media type. Therefore, test for true explicitly.

If the filter is running, a return value of true is ambiguous. The pin might accept a format change on the next media sample, without reconnecting; or it might need to reconnect. If the pin supports the IPinConnection interface, call the IPinConnection::DynamicQueryAccept method, which specifically tests whether the pin can accept the new type without reconnecting.

10.28.4   BaseFilter as DirectShowBaseFilterMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries base filter.
Notes: Lasterror is set.

10.28.5   ConnectedTo as DirectShowPinMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ConnectedTo method retrieves a pointer to the connected pin, if any.
Notes: Lasterror is set.

10.28.6   ConnectionMediaType as DirectShowMediaTypeMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ConnectionMediaType method retrieves the media type for the current pin connection, if any.
Notes: Lasterror is set.
Returns nil if not connected.

10.28.7   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.28.8   Direction as Integer

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The QueryDirection method gets the direction of the pin (input or output).
Notes: Lasterror is set.

10.28.9   Disconnect

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Disconnect method breaks the current pin connection.
Notes: The Filter Graph Manager calls this method when it disconnects two filters. Applications and filters should not call this method. Instead, call the DirectShowFilterGraphMBS.Disconnect method on the Filter Graph Manager.
Sets lasterror.

This method fails if the filter is paused or running. If the pin supports the PinConnection interface, call PinConnection DynamicDisconnect to disconnect the pin when the filter is paused or running.
This method does not disconnect the other pin in the pin connection.

10.28.10 Id as String

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Id method retrieves an identifier for the pin.
Notes: Returns a string containing the pin identifier.
Lasterror is set.

10.28.11 MediaTypes as DirectShowMediaTypeMBS()

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enumerates the pin's preferred media types.
Notes: Lasterror is set.

10.28.12 Name as String

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries name of the pin.
Notes: Lasterror is set.

10.28.13 PIN_CATEGORY_ANALOGVIDEOIN as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Input pin of the capture filter that takes analog and digitizes it.

10.28.14   PIN_CATEGORY_CAPTURE as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Capture pin.

10.28.15   PIN_CATEGORY_CC as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing closed captioning data from Line 21.

10.28.16   PIN_CATEGORY_EDS as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing Extended Data Services (Line 21, even fields).

10.28.17   PIN_CATEGORY_NABTS as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing North American Videotext Standard data.

10.28.18   PIN_CATEGORY_PREVIEW as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Preview pin.

10.28.19   PIN_CATEGORY_STILL as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin that provides a still image. The filter's capture pin must be connected before the still-image pin is connected.

10.28.20 PIN_CATEGORY_TELETEXT as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing teletext (a closed captioning variant).

10.28.21 PIN_CATEGORY_TIMECODE as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing timecode data.

10.28.22 PIN_CATEGORY_VBI as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin to be connected to the VBI Surface Allocator, the VBI surface allocator filter that is needed to allocate the correct video memory for things like closed captioning overlays in scenarios where a video port is being used. PCI, IEEE 1394, and USB scenarios do not use this filter.

10.28.23 PIN_CATEGORY_VIDEOPORT as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Video output pin to be connected to input pin zero on the Overlay Mixer.

10.28.24 PIN_CATEGORY_VIDEOPORT_VBI as DirectShowGUIDMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the DirectShow pin categories.
Notes: Pin providing vertical blanking interval data.

10.28.25   Properties

10.28.26   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IPin interface.
(Read and Write property)

10.28.27   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.28.28   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.28.29   Constants

Pin Direction

Constant	Value	Description
kPinDirectionInput	0	Input
kPinDirectionOutput	1	Output

10.29 class DirectShowPropertyBagMBS

10.29.1 class DirectShowPropertyBagMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Provides an object with a property bag in which the object can save its properties persistently.
Notes: used for Monikers to store properties like description and user friendly name.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- News from the MBS Xojo Plugins Version 22.0

- MBS Xojo Plugins, version 21.6pr1

- MBS Real Studio Plugins, version 12.1pr2

10.29.2 Methods

10.29.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.29.4 CountProperties as Integer

Plugin Version: 22.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the number of properties in the property bag.
Notes: Sets lasterror.
This fails if the underlaying object doesn't implement the IPropertyBag2 protocol.

10.29.5 Description as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience method to read description.
Notes: Lasterror is set.

10.29.6   DevicePath as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience method to read device path.
Notes: Lasterror is set.

10.29.7   FriendlyName as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Convenience method to read friendly name.
Notes: Lasterror is set.

10.29.8   PropertyName(index as Integer) as string

Plugin Version: 22.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries name of a property.
Notes: Sets lasterror.
This fails if the underlaying object doesn't implement the IPropertyBag2 protocol.

10.29.9   Read(name as string) as Variant

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads one of the properties into a variant.
Notes: As we only need 3 properties for DirectShowMonikerMBS.Properties, we have direct accessors.
Lasterror is set.

10.29.10   Properties

10.29.11   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IPropertyBag interface.
(Read and Write property)

10.29.12   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.29.13   LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.30 class DirectShowSampleGrabberMBS

10.30.1 class DirectShowSampleGrabberMBS

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a sample grabber.
Notes: The Sample Grabber filter provides a way to retrieve samples as they pass through the filter graph. It is a transform filter with one input pin and one output pin. It passes all samples downstream unchanged, so you can insert it into a filter graph without altering the data stream. Your application can then retrieve individual samples from the filter by calling methods on the ISampleGrabber interface.
If you want to retrieve samples without rendering the data, connect the Sample Grabber filter to the Null Renderer filter.
Blog Entries

- MBS Real Studio Plugins, version 12.3pr8

10.30.2 Methods

10.30.3 BaseFilter as DirectShowBaseFilterMBS

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the filter for this grabber.
Notes: So you can add it to a graph builder.

10.30.4 ConnectedMediaType as DirectShowBaseFilterMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetConnectedMediaType method retrieves the media type for the connection on the input pin of the Sample Grabber.
Notes: Lasterror is set.

10.30.5 Constructor

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new instance of the sample grabber.

10.30.6   Current as Picture

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Queries current picture.
Notes: This actually makes a copy of the internal buffer into a new picture object.
Lasterror is set.

10.30.7   Destructor

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.

10.30.8   SetOneShot(OneShot as boolean)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetOneShot method specifies whether the Sample Grabber filter halts after the filter receives a sample.

10.30.9   Properties

10.30.10   Handle as Integer

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an ISampleGrabber interface.
(Read and Write property)

10.30.11   Lasterror as Integer

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.30.12   LasterrorMessage as String

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.30.13   Events

10.30.14   NewFrame(Time as Double)

Plugin Version: 12.4, Platform: Windows, Targets: .

Function: The event for a new video frame.
Notes: time is the starting time of the sample, in seconds.
Use the Current method to get the picture.

10.31 class DirectShowVideoInfoHeader2MBS

10.31.1 class DirectShowVideoInfoHeader2MBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes the bitmap and color information for a video image, including interlace, copy protection, and pixel aspect ratio information.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407326(v=vs.85).aspx
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 13.4pr7

10.31.2 Methods

10.31.3 Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.31.4 Properties

10.31.5 AvgTimePerFrame as Int64

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The desired average display time of the video frames, in 100-nanosecond units.
Notes: The actual time per frame may be longer.
see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407326(v=vs.85).aspx
(Read and Write property)

10.31.6 BitErrorRate as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Data error rate, in bit errors per second.
Notes: (Read and Write property)

10.31.7 BitRate as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Approximate data rate of the video stream, in bits per second.
Notes: (Read and Write property)

10.31.8 ControlFlags as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Control flags.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407326(v=vs.85).aspx
(Read and Write property)

10.31.9 CopyProtectFlags as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Flag set with the AMCOPYPROTECT_RestrictDuplication value (0x00000001) to indicate that the duplication of the stream should be restricted.
Notes: If undefined, specify zero or else the connection will be rejected.
(Read and Write property)

10.31.10 Height as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bitmap height.
Notes: (Read and Write property)

10.31.11 InterlaceFlags as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Flags that specify how the video is interlaced.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407326(v=vs.85).aspx
(Read and Write property)

10.31.12 PictAspectRatioX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The X dimension of picture aspect ratio. For example, 16 for a 16-inch x 9-inch display.
Notes: (Read and Write property)

10.31.13 PictAspectRatioY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y dimension of picture aspect ratio. For example, 9 for a 16-inch x 9-inch display.
Notes: (Read and Write property)

10.31.14 SourceBottom as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.31.15 SourceLeft as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.31.16 SourceRight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.31.17   SourceTop as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.31.18   TargetBottom as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.31.19   TargetLeft as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.31.20   TargetRight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.31.21   TargetTop as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.31.22   Width as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bitmap width.
Notes: (Read and Write property)

10.32   class DirectShowVideoInfoHeaderMBS

10.32.1   class DirectShowVideoInfoHeaderMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes the bitmap and color information for a video image.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407325(v=vs.85).aspx
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 16.3pr3

- MBS Xojo / Real Studio Plugins, version 13.4pr7

10.32.2   Methods

10.32.3   Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.32.4   Properties

10.32.5   AvgTimePerFrame as Int64

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The desired average display time of the video frames, in 100-nanosecond units.
Notes: The actual time per frame may be longer.
see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407325(v=vs.85).aspx
(Read and Write property)

10.32.6   BitCount as Integer

Plugin Version: 16.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bitmap bit count value.
Notes: The number of bits-per-pixel. Determines the number of bits that define each pixel and the maximum number of colors in the bitmap. This member must be one of the following values.

Can be 0 for PNG/JPEG images, 1 for B/W, 4 for 16 colors, 8 for 256 colors, 16 for thousands of colors, 24 for millions of colors or 32 for millions of colors with alpha.
(Read and Write property)

10.32.7   BitErrorRate as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Data error rate, in bit errors per second.
Notes: (Read and Write property)

10.32.8   BitRate as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Approximate data rate of the video stream, in bits per second.
Notes: (Read and Write property)

10.32.9   BMIHeaderPtr as Ptr

Plugin Version: 16.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns pointer to BITMAPINFOHEADER structure.
Notes: You can use this to modify any field directly.
You need to know what you do and know the exact offset. (which can be different in 32-bit vs. 64-bit)
(Read only property)

10.32.10   Height as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bitmap height.
Notes: (Read and Write property)

10.32.11   SourceBottom as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.

(Read and Write property)

10.32.12   SourceLeft as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.32.13   SourceRight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.32.14   SourceTop as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the rectangle defining the source video window.
Notes: This rectangle can be a clipping rectangle, to select a portion of the source video stream.
(Read and Write property)

10.32.15   TargetBottom as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.32.16   TargetLeft as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.32.17 TargetRight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.32.18 TargetTop as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Part of the target rectangle for the destination video window.
Notes: (Read and Write property)

10.32.19 VideoInfoHeaderPtr as Ptr

Plugin Version: 16.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns pointer to VIDEOINFOHEADER structure.
Notes: You can use this to modify any field directly.
You need to know what you do and know the exact offset. (which can be different in 32-bit vs. 64-bit)
(Read only property)

10.32.20 Width as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bitmap width.
Notes: (Read and Write property)

10.33 class DirectShowVideoStreamConfigCapsMBS

10.33.1 class DirectShowVideoStreamConfigCapsMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Describes a range of video formats.
Notes: Video compression and video capture filters use this structure to describe what formats they can produce.
Microsoft deprecated this strucutre some time ago in favor of DirectShowMediaTypeMBS class.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd407352(v=vs.85).aspx
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 13.4pr7

10.33.2 Methods

10.33.3 Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.33.4 Properties

10.33.5 CropAlignX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Required horizontal alignment of the source rectangle.
Notes: Deprecated.
(Read only property)

10.33.6 CropAlignY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Required vertical alignment of the source rectangle.
Notes: Deprecated.

(Read only property)

10.33.7 CropGranularityX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Horizontal granularity of the source rectangle.
Notes: This value specifies the increments that are valid between MinCroppingSize and MaxCroppingSize.
Deprecated.
(Read only property)

10.33.8 CropGranularityY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Vertical granularity of the source rectangle.
Notes: This value specifies the increments that are valid between MinCroppingSize and MaxCroppingSize.
Note Deprecated.
(Read only property)

10.33.9 InputSizeHeight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Native size of the incoming video signal.
Notes: For a compressor, the size is taken from the input pin. For a capture filter, the size is the largest signal the filter can digitize with every pixel remaining unique.
Deprecated.
(Read only property)

10.33.10 InputSizeWidth as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Native size of the incoming video signal.
Notes: For a compressor, the size is taken from the input pin. For a capture filter, the size is the largest signal the filter can digitize with every pixel remaining unique.
Deprecated.
(Read only property)

10.33.11   MaxBitsPerSecond as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum data rate this pin can produce.
Notes: Deprecated
(Read only property)

10.33.12   MaxCroppingSizeHeight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Largest source rectangle allowed.
Notes: Deprecated.
(Read only property)

10.33.13   MaxCroppingSizeWidth as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Largest source rectangle allowed.
Notes: Deprecated.
(Read only property)

10.33.14   MaxFrameInterval as Int64

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum frame duration, in 100-nanosecond units.
Notes: This value applies only to capture filters.
(Read and Write property)

10.33.15   MaxOutputSizeHeight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum output size.
Notes: Deprecated.
(Read only property)

10.33.16 MaxOutputSizeWidth as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Maximum output size.
Notes: Deprecated.
(Read only property)

10.33.17 MinBitsPerSecond as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum data rate this pin can produce.
Notes: Deprecated.
(Read only property)

10.33.18 MinCroppingSizeHeight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Smallest source rectangle allowed.
Notes: The source rectangle is defined in the Source rectangle of the DirectShowVideoInfoHeader2MBS or DirectShowVideoInfoHeaderMBS classes.
Note Deprecated.
(Read only property)

10.33.19 MinCroppingSizeWidth as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Smallest source rectangle allowed.
Notes: The source rectangle is defined in the Source rectangle of the DirectShowVideoInfoHeader2MBS or DirectShowVideoInfoHeaderMBS classes.
Note Deprecated.
(Read only property)

10.33.20 MinFrameInterval as Int64

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The minimum frame duration, in 100-nanosecond units. This value applies only to capture filters.

Notes: (Read and Write property)

10.33.21   MinOutputSizeHeight as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum output size.
Notes: Deprecated.
(Read only property)

10.33.22   MinOutputSizeWidth as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum output size.
Notes: Deprecated.
(Read only property)

10.33.23   OutputGranularityX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Granularity of the output width.
Notes: This value specifies the increments that are valid between MinOutputSize and MaxOutputSize.
Deprecated.
(Read only property)

10.33.24   OutputGranularityY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Granularity of output height.
Notes: This value specifies the increments that are valid between MinOutputSize and MaxOutputSize.
Deprecated.
(Read only property)

10.33.25   ShrinkTapsX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates how well the filter can shrink the image horizontally.
Notes: Deprecated.

Value	Meaning
0	Does not support stretching/shrinking.
1	Uses pixel doubling (stretching) or eliminates pixels (shrinking)
2	Uses interpolation (2 taps)
3 and higher	Uses interpolation (>2 taps)

(Read only property)

10.33.26   ShrinkTapsY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates how well the filter can shrink the image vertically.
Notes: Deprecated.

Value	Meaning
0	Does not support stretching/shrinking.
1	Uses pixel doubling (stretching) or eliminates pixels (shrinking)
2	Uses interpolation (2 taps)
3 and higher	Uses interpolation (>2 taps)

(Read only property)

10.33.27   StretchTapsX as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates how well the filter can stretch the image horizontally.
Notes: Deprecated.

(Read only property)

Value	Meaning
0	Does not support stretching/shrinking.
1	Uses pixel doubling (stretching) or eliminates pixels (shrinking)
2	Uses interpolation (2 taps)
3 and higher	Uses interpolation (>2 taps)

10.33.28   StretchTapsY as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates how well the filter can stretch the image vertically.
Notes: Deprecated.

Value	Meaning
0	Does not support stretching/shrinking.
1	Uses pixel doubling (stretching) or eliminates pixels (shrinking)
2	Uses interpolation (2 taps)
3 and higher	Uses interpolation (>2 taps)

(Read only property)

10.33.29   VideoStandard as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The analog video standard supported.
Notes: The value is a bitwise combination of flags from the AnalogVideoStandard enumeration type, or zero.
(Read only property)

10.34 class DirectShowVideoWindowMBS

10.34.1 class DirectShowVideoWindowMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The direct show class for a video window.
Notes: The DirectShowVideoWindowMBS interface sets properties on the video window. Applications can use it to set the window owner, the position and dimensions of the window, and other properties.

The Video Renderer filter and the Filter Graph Manager both expose this interface. The Filter Graph Manager forwards all method calls to the Video Renderer. It also forwards certain window messages that the Video Renderer needs to receive, such as WM_DISPLAYCHANGE. Because the video window is usually a child of an application window, the filter would not otherwise receive these messages. Therefore it relies on the Filter Graph Manager to forward them.

In most cases, an application should query the Filter Graph Manager for this interface, and not call the filter directly, because of the messaging issue just described. However, if the filter graph has more than one Video Renderer, the Filter Graph Manager only communicates with one of them, selected arbitrarily. Therefore, if your application uses multiple video windows, use the DirectShowVideoWindowMBS interface directly on the filters. In that case, you must forward window messages to each Video Renderer instance, using the NotifyOwnerMessage method.

Properties set on a video renderer persist between successive connections and disconnections.

Error codes: If the video renderer filter is not connected to another filter, all methods return the error code VFW_E_NOT_CONNECTED. For the Filter Graph Manager's implementation, if the graph does not contain a video renderer filter, all methods return E_NOINTERFACE. Note that the Filter Graph Manager exposes the interface even when the graph does not contain a video renderer, so an application can query for the interface before it builds the graph.
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 21.6pr3
- MBS Xojo Plugins, version 21.0pr7

10.34.2 Methods

10.34.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor to create a new video window.

10.34.4   GetMaxIdealImageSize(byref width as Integer, byref height as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetMaxIdealImageSize method retrieves the maximum ideal image size for the video image.
Notes: Width: Receives the maximum ideal width, in pixels.
Height: Receives the maximum ideal height, in pixels.
Lasterror is set.

The maximum ideal size may differ from the native video size, because the video hardware might have specific stretching requirements.

10.34.5   GetMinIdealImageSize(byref width as Integer, byref height as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetMinIdealImageSize method retrieves the minimum ideal size for the video image.
Notes: Width: Receives the minimum ideal width, in pixels.
Height: Receives the minimum ideal height, in pixels.
Lasterror is set.

The maximum ideal size may differ from the native video size, because the video hardware might have specific stretching requirements.

10.34.6   GetRestorePosition(byref left as Integer, byref top as Integer, byref width as Integer, byref height as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetRestorePosition method retrieves the restored window position.
Notes: Left: Receives the x-coordinate, in pixels.
Top: Receives the y-coordinate, in pixels.
Width: Receives the width of the window, in pixels.
Height: Receives the height of the window, in pixels.
Lasterror is set.

If the video window is minimized or maximized, you can use this method to get the window's restored position.

10.34.7 GetWindowPosition(byref left as Integer, byref top as Integer, byref width as Integer, byref height as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetWindowPosition method retrieves the position of the video window.
Notes: Left: Receives the x-coordinate, in pixels.
Top: Receives the y-coordinate, in pixels.
Width: Receives the width of the window, in pixels.
Height: Receives the height of the window, in pixels.

Lasterror is set.
This method has the same effect as querying left, top, width and height methods.

10.34.8 HideCursor(hide as boolean)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The HideCursor method shows or hides the cursor when the mouse is positioned over the video window.
Notes: Hide: Whether to hide or show the cursor.
Lasterror is set.

10.34.9 IsCursorHidden as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IsCursorHidden method queries whether the cursor is hidden.
Notes: Lasterror is set.

10.34.10 SetWindowForeground(Focus as Boolean)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetWindowForeground method places the video window at the top of the Z order.
Notes: Focus: whether to give the window focus.
Lasterror is set.

10.34.11 SetWindowPosition(left as Integer, top as Integer, width as Integer, height as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetWindowPosition method sets the position of the video window.
Notes: Left: The x-coordinate, in pixels.
Top: The y-coordinate, in pixels.
Width: The width, in pixels.
Height: The height, in pixels.
Lasterror is set.

This method has the same effect as setting the Left, Top, Width, and Height methods.
If resizing the window to the specified dimensions is impossible, this method modifies the window's size and location to make the window fit. Call the GetWindowPosition method to determine the result.

10.34.12 Properties

10.34.13 Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Points to an IVideoWindow interface.
(Read and Write property)

10.34.14 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Please check function documentation and also LastErrorMessage property for a human readable error message.
(Read and Write property)

10.34.15 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The formatted error message for the last error.
Notes: (Read and Write property)

10.34.16   MessageDrain as Variant

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The window to receive mouse and keyboard messages from the video window.
Notes: Lasterror is set.
This method enables an application to respond to mouse and keyboard events generated within the video window.
If Drain is non-nil, the video renderer forwards certain messages to the specified window, using the PostMessage function. Which messages are forwarded might depend on the video renderer in use.

Can reference a Window or DesktopWindow object.
(Read and Write property)

10.34.17   messageDrainControl as Variant

Plugin Version: 21.0, Platform: Windows, Targets: Desktop only.

Function: The control to receive mouse and keyboard messages from the video window.
Notes: Lasterror is set.
This method enables an application to respond to mouse and keyboard events generated within the video window.
If Drain is non-nil, the video renderer forwards certain messages to the specified window, using the PostMessage function. Which messages are forwarded might depend on the video renderer in use.

Can reference a Control or DesktopControl object.
(Read and Write property)

10.34.18   Owner as Variant

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The parent window for the video window.
Notes: Lasterror is set.
Use this method to display videos in a compound document. This method changes the parent of the video window and sets the WS_CHILD style for the video window.
Reset the owner to nil before releasing the Filter Graph Manager. Otherwise, messages will continue to be sent to this window and errors will likely occur when the application is terminated.

Can reference a Window or DesktopWindow object.
(Read and Write property)

10.34.19   OwnerControl as Variant

Plugin Version: 21.0, Platform: Windows, Targets: Desktop only.

Function: The parent control for the video window.
Notes: Lasterror is set.
Use this method to display videos in a compound document. This method changes the parent of the video window and sets the WS_CHILD style for the video window.
Reset the owner to nil before releasing the Filter Graph Manager. Otherwise, messages will continue to be sent to this window and errors will likely occur when the application is terminated.

Can reference a Control or DesktopControl object.
(Read and Write property)

10.34.20   AutoShow as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the video renderer automatically shows the video window when it receives video data.
Notes: Lasterror is set.
By default, when the filter graph changes state to paused or running, the video renderer shows the video window and moves it to the foreground. If the user closes the window, it will not automatically reappear.
(Read and Write computed property)

10.34.21   BackgroundPalette as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the video window realizes its palette in the background.
Notes: Lasterror is set.
If BackgroundPalette is true and the video image requires a palette, the video renderer will realize that palette in the background. Any colors that the palette uses will change to their closest match in the display palette prior to drawing. This ensures that an application will not have its palette disturbed. However, it imposes severe performance penalties on the video.
(Read and Write computed property)

10.34.22   BorderColor as color

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The color that appears around the edges of the destination rectangle.
Notes: Lasterror is set.
If the destination rectangle is smaller than the client area of the video window, a border is exposed around

the edges of the video. The default color is black. Use this method to override the default color. If a palette is in use, a nonsystem color is converted to its closest match.
(Read and Write computed property)

10.34.23   Caption as string

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The video window caption.
Notes: Lasterror is set.
(Read and Write computed property)

10.34.24   FullScreenMode as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enables or disables full-screen video rendering.
Notes: Set to true to switch to full-screen mode. Set to false to disable full-screen mode. (Default.)
Lasterror is set.

Depending on the video renderer, the switch to full-screen mode may not be visible until the application runs or pauses the graph. In full-screen mode, if the user switches away from the application (for example, using ALT + TAB), the Filter Graph Manager sends an EC_FULLSCREEN_LOST event.
The following remarks describe how the Filter Graph Manager implements full-screen mode. Application developers can probably ignore this information, but it may be useful if you are writing a custom video renderer.
When an application switches to full-screen mode, the Filter Graph Manager searches for a video renderer that will function most efficiently. In order of preference, these are:

- Any video renderer in the filter graph that natively supports full-screen mode.

- Any video renderer in the filter graph that can stretch the video to full-screen without a significant performance cost.

- The Full Screen Renderer filter.

- Any video renderer in the filter graph that supports DirectShowVideoWindowMBS.

For the first option, the Filter Graph Manager sets FullScreenMode on every video renderer in the graph. Most renderers return E_NOTIMPL, indicating the filter does not natively support full-screen mode. If any renderer returns a value not equal to E_NOTIMPL, the Filter Graph Manager uses that one.
For the second option, the Filter Graph Manager calls GetMaxIdealImageSize and GetMinIdealImageSize on every video renderer in the graph. If the size of the display falls within the filter's reported range, it

indicates that the filter can stretch the video without a significant performance cost.

Note If the graph is stopped, the Filter Graph Manager pauses each renderer before calling these methods. This gives the renderer an opportunity to initialize any resources it needs, because many renderers cannot determine these values while they are stopped.

Except on older hardware, the second option will generally succeed. The third option is to use the Full Screen Renderer filter, adding it to the graph if necessary. The fourth option is simply to find the first renderer in the graph that supports DirectShowVideoWindowMBS, and stretch the video regardless of performance. (Read and Write computed property)

10.34.25   Height as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the video window.
Notes: Lasterror is set.
(Read and Write computed property)

10.34.26   Left as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The x-coordinate of the video window.
Notes: Lasterror is set.
(Read and Write computed property)

10.34.27   Top as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The y-coordinate of the video window.
Notes: Lasterror is set.
(Read and Write computed property)

10.34.28   Visible as Boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Shows or hides the video window.
Notes: Set to true to show the window or set to false to hide the window.
Lasterror is set.

(Read and Write computed property)

10.34.29   Width as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the video window.
Notes: (Read and Write computed property)

10.34.30   WindowState as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Shows, hides, minimizes, or maximizes the video window.
Notes: Lasterror is set.

See MSDN page for ShowWindow for SW_* constants:
http://msdn.microsoft.com/en-us/library/windows/desktop/ms633548(v=vs.85).aspx
(Read and Write computed property)

10.34.31   WindowStyle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The window styles on the video window.
Notes: Lasterror is set.
See MSDN for SetWindowLong for details on the WS_* constants.
(Read and Write computed property)

10.34.32   WindowStyleEx as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The window extended styles on the video window.
Notes: Lasterror is set.
See MSDN for SetWindowLong for details on the WS_EX_* constants.
(Read and Write computed property)

10.34.33   Constants

Window Show Modes

Constant	Value	Description
SW_FORCEMINIMIZE	11	
SW_HIDE	0	
SW_MAXIMIZE	3	
SW_MINIMIZE	6	
SW_NORMAL	1	
SW_RESTORE	9	
SW_SHOW	5	
SW_SHOWDEFAULT	10	
SW_SHOWMAXIMIZED	3	
SW_SHOWMINIMIZED	2	
SW_SHOWMINNOACTIVE	7	
SW_SHOWNA	8	
SW_SHOWNOACTIVATE	4	
SW_SHOWNORMAL	1	

Windows Flags

Constant	Value	Description
WS_BORDER	&h00800000	
WS_CAPTION	&h00C00000	
WS_CHILD	&h40000000	
WS_CHILDWINDOW	&h40000000	
WS_CLIPCHILDREN	&h02000000	
WS_CLIPSIBLINGS	&h04000000	
WS_DISABLED	&h08000000	
WS_DLGFRAME	&h00400000	
WS_GROUP	&h00020000	
WS_HSCROLL	&h00100000	
WS_ICONIC	&h20000000	
WS_MAXIMIZE	&h01000000	
WS_MAXIMIZEBOX	&h00010000	
WS_MINIMIZE	&h20000000	
WS_MINIMIZEBOX	&h00020000	
WS_OVERLAPPED	&h00000000	
WS_OVERLAPPEDWINDOW	&h00CF0000	
WS_POPUP	&h80000000	
WS_POPUPWINDOW	&h80880000	
WS_SIZEBOX	&h00040000	
WS_SYSMENU	&h00080000	
WS_TABSTOP	&h00010000	
WS_THICKFRAME	&h00040000	
WS_TILED	&h00000000	
WS_TILEDWINDOW	&h00CF0000	
WS_VISIBLE	&h10000000	
WS_VSCROLL	&h00200000	

Windows Extended Flags

Constant	Value	Description
WS_EX_ACCEPTFILES	&h00000010	
WS_EX_APPWINDOW	&h00040000	
WS_EX_CLIENTEDGE	&h00000200	
WS_EX_CONTEXTHELP	&h00000400	
WS_EX_CONTROLPARENT	&h00010000	
WS_EX_DLGMODALFRAME	&h00000001	
WS_EX_LAYERED	&h00080000	
WS_EX_LAYOUTRTL	&h00400000	
WS_EX_LEFT	&h00000000	
WS_EX_LEFTSCROLLBAR	&h00004000	
WS_EX_LTRREADING	&h00000000	
WS_EX_MDICHILD	&h00000040	
WS_EX_NOACTIVATE	&h08000000	
WS_EX_NOINHERITLAYOUT	&h00100000	
WS_EX_NOPARENTNOTIFY	&h00000004	
WS_EX_OVERLAPPEDWINDOW	&h00000300	
WS_EX_PALETTEWINDOW	&h00000188	
WS_EX_RIGHT	&h00001000	
WS_EX_RIGHTSCROLLBAR	&h00000000	
WS_EX_RTLREADING	&h00002000	
WS_EX_STATICEDGE	&h00020000	
WS_EX_TOOLWINDOW	&h00000080	
WS_EX_TOPMOST	&h00000008	
WS_EX_TRANSPARENT	&h00000020	
WS_EX_WINDOWEDGE	&h00000100	

10.35 class DirectShowWaveFormatMBS

10.35.1 class DirectShowWaveFormatMBS

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Defines the format of waveform-audio data.
Notes: Only format information common to all waveform-audio data formats is included in this structure. For formats that require additional information, this structure is included as the first member in another structure, along with the additional information.
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 13.4pr7

10.35.2 Methods

10.35.3 Constructor

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

10.35.4 Properties

10.35.5 AvgBytesPerSec as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Required average data-transfer rate, in bytes per second, for the format tag.
Notes: If wFormatTag is WAVE_FORMAT_PCM, nAvgBytesPerSec must equal nSamplesPerSec x nBlockAlign. For non-PCM formats, this member must be computed according to the manufacturer's specification of the format tag.
(Read and Write property)

10.35.6 BitsPerSample as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Bits per sample for the FormatTag format type.
Notes: If wFormatTag is WAVE_FORMAT_PCM, then BitsPerSample should be equal to 8 or 16. For non-PCM formats, this member must be set according to the manufacturer's specification of the format tag. If FormatTag is WAVE_FORMAT_EXTENSIBLE, this value can be any integer multiple of 8.

Some compression schemes do not define a value for BitsPerSample, so this member can be zero.
(Read and Write property)

10.35.7   BlockAlign as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Block alignment, in bytes.
Notes: The block alignment is the minimum atomic unit of data for the wFormatTag format type. If wFormatTag is WAVE_FORMAT_PCM, nBlockAlign must equal (nChannels √ó wBitsPerSample) / 8. For non-PCM formats, this member must be computed according to the manufacturer's specification of the format tag.
Software must process a multiple of nBlockAlign bytes of data at a time. Data written to and read from a device must always start at the beginning of a block. For example, it is illegal to start playback of PCM data in the middle of a sample (that is, on a non-block-aligned boundary).
(Read and Write property)

10.35.8   Channels as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of channels in the waveform-audio data.
Notes: Monaural data uses one channel and stereo data uses two channels.
(Read and Write property)

10.35.9   Data as Ptr

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pointer to raw data of WAVEFORMATEX structure.
Notes: (Read only property)

10.35.10   FormatTag as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Waveform-audio format type.
Notes: Format tags are registered with Microsoft Corporation for many compression algorithms. A complete list of format tags can be found in the Mmreg.h header file. For one- or two-channel Pulse Code Modulation (PCM) data, this value should be WAVE_FORMAT_PCM.

- If FormatTag equals WAVE_FORMAT_EXTENSIBLE, the structure is interpreted as a WAVEFOR-MATEXTENSIBLE structure.

- If FormatTag equals WAVE_FORMAT_MPEG, the structure is interpreted as an MPEG1WAVE-FORMAT structure.

- If FormatTag equals WAVE_FORMAT_MPEGLAYER3, the structure is interpreted as an MPEGLAYER3WAVE-FORMAT structure.

Before reinterpreting a WAVEFORMATEX structure as one of these extended structures, verify that the actual structure size is sufficiently large and that the cbSize member indicates a valid size.

Please contact MBS if you need the above structures as classes in Xojo.
(Read and Write property)

10.35.11   SamplesPerSec as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sample rate, in samples per second (hertz).
Notes: If FormatTag is WAVE_FORMAT_PCM, then common values for SamplesPerSec are 8.0 kHz, 11.025 kHz, 22.05 kHz, and 44.1 kHz. For non-PCM formats, this member must be computed according to the manufacturer's specification of the format tag.
(Read and Write property)

10.35.12   Size as Integer

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Size, in bytes, of extra format information appended to the end of the WAVEFORMATEX structure.
Notes: This information can be used by non-PCM formats to store extra attributes for the wFormatTag. If no extra information is required by the wFormatTag, this member must be set to zero. For WAVE_FOR-MAT_PCM formats (and only WAVE_FORMAT_PCM formats), this member is ignored. However it is still recommended to set the value.
(Read and Write property)

Chapter 11

DiscRecording

11.1 class WindowsBurnMBS

11.1.1 class WindowsBurnMBS

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Exposes methods that determine whether a system has hardware for writing to CD, the drive letter of a CD writer device, and programmatically initiate a CD writing session.

11.1.2 Methods

11.1.3 CDBurn

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Instructs data to be copied from the staging area to a writable CD.
Notes: The staging area has a default location of %userprofile%\Local Settings\Application Data\Microsoft\CD Burning. Its actual path can be retrieved through WindowsBurnAreaFolderMBS.

This method returns when the CD is done or thre user cancelled.
See also:

11.1.4   CDBurn(hostwindow as DesktopWindow)

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

Function: Instructs data to be copied from the staging area to a writable CD.
Notes: Hostwindow: parent window of the user interface (UI)

The staging area has a default location of %userprofile%\Local Settings\Application Data\Microsoft\CD Burning. Its actual path can be retrieved through WindowsBurnAreaFolderMBS.

This method returns when the CD is done or thre user cancelled.
See also:

11.1.5   CDBurn(hostwindow as window)

Plugin Version: 8.5, Platform: Windows, Targets: Desktop only.

Function: Instructs data to be copied from the staging area to a writable CD.
Notes: Hostwindow: parent window of the user interface (UI)

The staging area has a default location of %userprofile%\Local Settings\Application Data\Microsoft\CD Burning. Its actual path can be retrieved through WindowsBurnAreaFolderMBS.

This method returns when the CD is done or thre user cancelled.
See also:

11.1.6   HasRecordableDrive as boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scans the system for a CD drive with write-capability, returning true if one is found.
Notes: This search does not rely on the state of the Enable cd writing on this drive option found on the drive's property sheet. Instead, the determination is based on IMAPI.

11.1.7   RecorderDriveLetter as string

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the drive letter of a CD drive that has been marked as write-enabled.
Notes: The drive whose letter designation is returned by this method is the drive that has the Enable cd writing on this drive option selected. This option is found on the drive's property sheet. Only one drive on a system can have this option selected.

If a recordable CD drive is present but that option has been deselected, the method will return ane empty string.

11.1.8   Properties

11.1.9   Available as boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the Shell extension on Windows for CD Burning is available.
Notes: Available should be true on Windows XP and newer.
(Read only property)

11.1.10   Lasterror as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code reported by one of the methods.
Notes: (Read and Write property)

Chapter 12

Drag & Drop

12.1   class WinDataObjectMBS

12.1.1   class WinDataObjectMBS

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a data object for Drag and Drop.
Example:

```
// take some picture
dim p as Picture = LogoMBS(500)

// create data object
dim w as new WinDataObjectMBS(p)
```

Notes: While the drag classes compile for Web Edition, they run on the server, so they have no effect on the client browser!
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.4

- MBS Xojo Plugins, version 20.4pr3

- MBS Xojo Plugins, version 20.3pr9

- MBS Xojo plug-ins in version 16.0

- MBS Xojo / Real Studio Plugins, version 16.0pr8

- More Drag & Drop on Windows

- MBS Xojo / Real Studio Plugins, version 15.5pr3

381

- MBS Real Studio Plugins, version 11.2pr4

- MBS REALbasic Plugins, version 10.6pr6

- MBS REALbasic Plugins, version 10.6pr5

Xojo Developer Magazine

- 18.6, page 10: News

12.1.2   Methods

12.1.3   AddDragImage(pic as picture, width as Integer, height as Integer, x as Integer, y as Integer)

Plugin Version: 11.0, Platform: Windows, Targets: Desktop only.

Function: Adds a drag image to the data object.
Notes: pic: the picture to use.
width and height: The size of the picture.
x and y: The location of the cursor within the drag image. The point should contain the offset from the upper-left corner of the drag image to the location of the cursor.

Requires Windows 2000 Professional with SP3, Windows XP.
On success the HelperHandle property is not zero.

Turn off antialiasing when drawing text. Otherwise, artifacts could occur at the edges, between the text color and the color key.

This function takes the picture (and it's mask) and turns it in a nice drag picture. This includes applying the mask and passing black for the background color. Dark colors which should be transparent will be made lighter.
See also:

- 12.1.4 AddDragImage(pic as picture, width as Integer, height as Integer, x as Integer, y as Integer, ImageBackgroundColor as color)   382

12.1.4   AddDragImage(pic as picture, width as Integer, height as Integer, x as Integer, y as Integer, ImageBackgroundColor as color)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop only.

Function: Adds a drag image to the data object.
Notes: pic: the picture to use.

width and height: The size of the picture.

x and y: The location of the cursor within the drag image. The point should contain the offset from the upper-left corner of the drag image to the location of the cursor.

ImageBackgroundColor: The color used by the control to fill the background of the drag image.

Requires Windows 2000 Professional with SP3, Windows XP.
On success the HelperHandle property is not zero.

Turn off antialiasing when drawing text. Otherwise, artifacts could occur at the edges, between the text color and the color key.
See also:

- 12.1.3 AddDragImage(pic as picture, width as Integer, height as Integer, x as Integer, y as Integer) 382

12.1.5 AddFiles(files() as folderitem)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds file references in the format the explorer can understand them.
See also:

- 12.1.6 AddFiles(pathes() as string) 383

12.1.6 AddFiles(pathes() as string)

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds file pathes in the format the explorer can understand them.
Notes: Folder pathes should have no backslash on the end.
See also:

- 12.1.5 AddFiles(files() as folderitem) 383

12.1.7 AddPicture(pic as picture)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop only.

Function: Adds a picture to the data object.

12.1.8 AddRaw(format as Integer, data as string)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds raw data to the data object.
Notes: Depending of the format you may need to add chr(0) on the end.

12.1.9   AddText(text as string)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds the text to the data object.

12.1.10   Constructor

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an empty data object.
See also:

12.1.11   Constructor(files() as folderitem)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a data object and adds the files.
See also:

12.1.12   Constructor(pic as picture)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop only.

Function: Creates a new data object and adds the given picture.
Example:

```
// take some picture
dim p as Picture = LogoMBS(500)
```

// create data object
dim w as new WinDataObjectMBS(p)

See also:

12.1.13   Constructor(text as string)

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new data object and adds the given text.
Example:

// create data object with text
dim w as new WinDataObjectMBS("Hello World")

See also:

12.1.14   Formats as String()

Plugin Version: 16.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries list of formats in the data object.
Notes: Helps for debugging to see what is inside.

12.1.15   GetFileContents(index as Integer) as string

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries file content for the file with the given index.
Notes: Use index from WindowsFileDescriptorMBS.index property.

This works for files up to a few hundred megabytes in size. For larger files we will have to change plugin if you want to receive those.

Only accepts data provided as blob.
See also:

12.1.16   GetFileContents(index as integer, byref IsPath as boolean) as string

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries file content for the file with the given index.
Notes: Use index from WindowsFileDescriptorMBS.index property.

This works for files up to a few hundred megabytes in size. For larger files we will have to change plugin if you want to receive those.

IsPath is set to true, when the returned data is a file path instead of data itself.
See also:

12.1.17   GetFileDescriptors as WindowsFileDescriptorMBS()

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries file descriptors.
Example:

dim dataObject as WinDataObjectMBS // your data object

dim des(-1) as WindowsFileDescriptorMBS = dataObject.GetFileDescriptors

for each d as WindowsFileDescriptorMBS in des
// we got file descriptions. Some metadata and the data. No path.
dim data as string = dataObject.GetFileContents(0)
msgbox "File """+d.FileName+""" with "+str(lenb(data))+" bytes data."
next

Notes: Result array is empty on any error.

12.1.18 GetFileName as string

Plugin Version: 16.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries filename from data object.
Notes: Depending on how the other application works, the drag data object may have either file descriptors, a file path or path strings.

The file name is valid at least as long as this object is alive.

12.1.19 GetPaths as folderitem()

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the paths in the data object.
Example:

dim dataObject as WinDataObjectMBS // your data object

dim files(-1) as FolderItem = dataObject.GetPaths

for each f as FolderItem in files
// we got a file you can use like any other file (e.g. copy)
listbox1.AddRow "Path """+f.NativePath+""""
next

Notes: Checks for a CF_HDROP type. May return one or more folderitems.

12.1.20 GetPathStrings as string()

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the paths in the data object.
Notes: Checks for a CF_HDROP type. May return one or more folderitems.

12.1.21 GetPicture as picture

Plugin Version: 10.5, Platform: Windows, Targets: Desktop only.

Function: Queries the picture from the data object.
Notes: Supports CF_BITMAP/TYMED_GDI.

12.1.22   GetRaw(format as Integer) as string

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the raw data for the given type.

12.1.23   GetText as string

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the text from the data object.
Notes: Returns unicode or ANSI text depending on what is available. Unicode is preferred.

12.1.24   HasFileDescriptors as boolean

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether this data object contains file descriptors.
Notes: Returns true if a path is found.
Checks for CF_FILEGROUPDESCRIPTOR.

Windows uses File Descriptors and FileContents for drag and drop operations where the data is not stored in a file. You get the descriptors and if you need you can get the data which is delivered just in time.

12.1.25   HasFileName as boolean

Plugin Version: 16.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks to see if a filename entry is in the data object.

12.1.26   HasPaths as boolean

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether this data object contains file paths.
Notes: Returns true if a path is found.
Checks for CF_HDROP.

12.1.27 HasPicture as boolean

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks if a picture is part of this data object.
Notes: Returns true if a picture is found.
Checks for CF_BITMAP.

12.1.28 HasRaw(format as Integer) as boolean

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether this data object contains data in the given format.

12.1.29 HasText as boolean

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether this data object contains Text or UnicodeText.
Notes: Returns true if text is found.
Checks for CF_UNICODETEXT and CF_TEXT.

12.1.30 Properties

12.1.31 DragImage as Picture

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The drag image.
Notes: Used to show user what he is dragging.
(Read only property)

12.1.32 Handle as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

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

12.1.33   HelperHandle as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal helper object to handle the drag image.
Notes: This value is not zero if the AddDragImage call was successful.
(Read and Write property)

12.1.34   Lasterror as Integer

Plugin Version: 16.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

12.1.35   Constants

Constants

Constant	Value	Description
CF_BITMAP	2	One of the constants for the data types.
CF_DIB	8	One of the constants for the data types.
CF_DIBV5	17	One of the constants for the data types.
CF_DIF	5	One of the constants for the data types.
CF_ENHMETAFILE	14	One of the constants for the data types.
CF_HDROP	15	One of the constants for the data types.
CF_LOCALE	16	One of the constants for the data types.
CF_METAFILEPICT	3	One of the constants for the data types.
CF_OEMTEXT	7	One of the constants for the data types.
CF_PALETTE	9	One of the constants for the data types.
CF_PENDATA	10	One of the constants for the data types.
CF_RIFF	11	One of the constants for the data types.
CF_SYLK	4	One of the constants for the data types.
CF_TEXT	1	One of the constants for the data types.
CF_TIFF	6	One of the constants for the data types.
CF_UNICODETEXT	13	One of the constants for the data types.
CF_WAVE	12	One of the constants for the data types.

12.2 class WindowsDragSourceMBS

12.2.1 class WindowsDragSourceMBS

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for Drag and Drop on Windows to create a drag source.
Notes: The WindowsDragSourceMBS class is one of the class you implement to provide drag-and-drop operations in your application. It contains methods used in any application used as a data source in a drag-and-drop operation. The data source application in a drag-and-drop operation is responsible for:

- Determining the data being dragged based on the user's selection.

- Initiating the drag-and-drop operation based on the user's mouse actions.

- Generating some of the visual feedback during the drag-and-drop operation, such as setting the cursor and highlighting the data selected for the drag-and-drop operation.

- Canceling or completing the drag-and-drop operation based on the user's mouse actions.

- Performing any action on the original data caused by the drop operation, such as deleting the data on a drag move.

WindowsDragSourceMBS contains the events for generating visual feedback to the end user and for canceling or completing the drag-and-drop operation.

When To Implement

Implement WindowsDragSourceMBS if you are developing a container or server application that can act as a data source for a drag-and-drop operation. The WindowsDragSourceMBS interface is only required during the drag-and-drop operation.

If you implement the WindowsDragSourceMBS class, you must also implement the DataObjectMBS class on the same object to represent the data being transferred.

You can use the same implementation of DataObjectMBS for drag-and-drop data as for the data object offered to the clipboard. After you have implemented clipboard operations in your application, you can add drag-and-drop operations with only a little extra work.

When To Use

You do not usually call the WindowsDragSourceMBS methods directly. Instead, your data source calls the DoDragDrop function when it detects that the user has initiated a drag-and-drop operation. Then, DoDragDrop calls the WindowsDragSourceMBS methods during the drag-and-drop operation.

For example, DoDragDrop calls WindowsDragSourceMBS.GiveFeedback when you need to change the cursor shape or when you need to provide some other visual feedback. DoDragDrop calls WindowsDragSourceMBS.QueryContinueDrag when there is a change in the mouse button state to determine if the drag-and-drop operation was canceled or completed.

While the drag classes compile for Web Edition, they run on the server, so they have no effect on the client browser!
Blog Entries

- MBS REALbasic Plugins, version 10.5pr5

12.2.2   Methods

12.2.3   DoDragDrop(dataObject as WinDataObjectMBS, OKEffect as Integer, byref Effect as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Carries out an OLE drag and drop operation.
Notes: Parameters:

dataObject: The WinDataObjectMBS data object that contains the data being dragged.
OKEffects: Effects the source allows in the OLE drag-and-drop operation. Most significant is whether it permits a move. The OKEffect and Effect parameters obtain values from the DROPEFFECT* constants.
Effect: Pointer to a value that indicates how the OLE drag-and-drop operation affected the source data. The pdwEffect parameter is set only if the operation is not canceled.

This function returns S_OK on success. Other possible values include the following.

Return code	Description
DRAGDROP_S_DROP	The OLE drag-and-drop operation was successful.
DRAGDROP_S_CANCEL	The OLE drag-and-drop operation was canceled.
E_UNSPEC	Unexpected error occurred.

Remarks

If you are developing an application that can act as a data source for an OLE drag-and-drop operation, you must call DoDragDrop when you detect that the user has started an OLE drag-and-drop operation.

The DoDragDrop function enters a loop in which it calls various methods in the WindowsDragSourceMBS and WindowsDropTargetMBS interfaces. (For a successful drag-and-drop operation, the application acting as the data source must also implement WindowsDragSourceMBS, while the target application must imple-

ment WindowsDropTargetMBS.)

The DoDragDrop function determines the window under the current cursor location. It then checks to see if this window is a valid drop target.

If the window is a valid drop target, DoDragDrop calls WindowsDropTargetMBS.DragEnter. This method supplies an effect code indicating what would happen if the drop actually occurred. For a list of valid drop effects, see the DROPEFFECT* constants.

DoDragDrop calls WindowsDragSourceMBS.GiveFeedback with the effect code so that the drop source interface can provide appropriate visual feedback to the user.

DoDragDrop tracks mouse cursor movements and changes in the keyboard or mouse button state.

If the user moves out of a window, DoDragDrop calls WindowsDropTargetMBS.DragLeave.

If the mouse enters another window, DoDragDrop determines if that window is a valid drop target and then calls WindowsDropTargetMBS.DragEnter for that window.

If the mouse moves but stays within the same window, DoDragDrop calls WindowsDropTargetMBS.DragOver.

If there is a change in the keyboard or mouse button state, DoDragDrop calls WindowsDragSourceMBS.QueryContinueDrag and determines whether to continue the drag, to drop the data, or to cancel the operation based on the return value.

If the return value is S_OK, DoDragDrop first calls WindowsDropTargetMBS.DragOver to continue the operation. This method returns a new effect value and DoDragDrop then calls WindowsDragSourceMBS.GiveFeedback with the new effect so appropriate visual feedback can be set. For a list of valid drop effects, see the DROPEFFECT constants. WindowsDropTargetMBS.DragOver and WindowsDragSourceMBS.GiveFeedback are paired so that as the mouse moves across the drop target, the user is given the most up-to-date feedback on the mouse's position.

If the return value is DRAGDROP_S_DROP, DoDragDrop calls WindowsDropTargetMBS.Drop. The DoDragDrop function returns the last effect code to the source, so the source application can perform the appropriate operation on the source data, for example, cut the data if the operation was a move.

If the return value is DRAGDROP_S_CANCEL, the DoDragDrop function calls WindowsDropTargetMBS.DragLeave.

12.2.4 Properties

12.2.5 Handle as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

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

12.2.6 Events

12.2.7 GiveFeedback(Effect as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Enables a source application to give visual feedback to the end user during a drag-and-drop operation by providing the DoDragDrop function with an enumeration value specifying the visual effect.
Notes: Effect: The drop effect value returned by the most recent call to WindowsDropTargetMBS.DragEnter, WindowsDropTargetMBS.DragOver, or WindowsDropTargetMBS.DragLeave.

Return S_OK on success. Return DRAGDROP_S_USEDEFAULTCURSORS to indicate successful completion of the method, and requests OLE to update the cursor using the OLE-provided default cursors.

When your application detects that the user has started a drag-and-drop operation, it should call the DoDragDrop function. DoDragDrop enters a loop, calling WindowsDropTargetMBS.DragEnter when the mouse first enters a drop target window, WindowsDropTargetMBS.DragOver when the mouse changes its position within the target window, and WindowsDropTargetMBS.DragLeave when the mouse leaves the target window.

For every call to either WindowsDropTargetMBS.DragEnter or WindowsDropTargetMBS.DragOver, DoDragDrop calls WindowsDropTargetMBS.GiveFeedback, passing it the drop effect value returned from the drop target call.

DoDragDrop calls WindowsDropTargetMBS.DragLeave when the mouse has left the target window. Then, DoDragDrop calls WindowsDropTargetMBS.GiveFeedback and passes the DROPEFFECT_NONE value in the dwEffect parameter.

The Effect parameter can include DROPEFFECT_SCROLL, indicating that the source should put up the drag-scrolling variation of the appropriate pointer.

Notes to Implementers

This function is called frequently during the DoDragDrop loop, so you can gain performance advantages if you optimize your implementation as much as possible.

GiveFeedback is responsible for changing the cursor shape or for changing the highlighted source based on the value of the dwEffect parameter. If you are using default cursors, you can return DRAGDROP_S_USEDE-FAULTCURSORS, which causes OLE to update the cursor for you, using its defaults.

12.2.8   QueryContinueDrag(EscapePressed as boolean, KeyState as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Determines whether a drag-and-drop operation should be continued, canceled, or completed. You do not call this method directly.
Notes: The OLE DoDragDrop function calls this method during a drag-and-drop operation.

EscapePressed: Indicates whether the Esc key has been pressed since the previous call to QueryContinue-Drag or to DoDragDrop if this is the first call to QueryContinueDrag. A true value indicates the end user has pressed the escape key; a false value indicates it has not been pressed.

KeyState: The current state of the keyboard modifier keys on the keyboard. Possible values can be a combination of any of the flags MK_CONTROL, MK_SHIFT, MK_ALT, MK_BUTTON, MK_LBUTTON, MK_MBUTTON, and MK_RBUTTON.

This event can return the following values.

Return code	Description
S_OK	The drag operation should continue. This result occurs if no errors are detected, the mouse button starting the drag-and-drop operation has not been released, and the Esc key has not been detected.
DRAGDROP_S_DROP	The drop operation should occur completing the drag operation. This result occurs if KeyState indicates that the key that started the drag-and-drop operation has been released.
DRAGDROP_S_CANCEL	The drag operation should be canceled with no drop operation occurring. This result occurs if EscapePressed is true, indicating the Esc key has been pressed.

The DoDragDrop function calls QueryContinueDrag whenever it detects a change in the keyboard or mouse button state during a drag-and-drop operation. QueryContinueDrag must determine whether the drag-and-drop operation should be continued, canceled, or completed based on the contents of the parameters KeyState and EscapePressed.

12.2.9   Constants

Constants

Constant	Value	Description
DRAGDROP_S_CANCEL	&H00040101	One of the OLE error codes.
		Drag and Drop was cancelled.
DRAGDROP_S_DROP	&h00040100	One of the OLE error codes.
		Do the drop operation.
DRAGDROP_S_USEDEFAULTCURSORS	&h00040102	One of the OLE error codes.
DROPEFFECT_COPY	1	One of the drop effect constants.
		Drop results in a copy. The original data is untouched by t
		Your application should always mask drop effect constants
		ibility with future implementations. Presently, only some
		in a drop effect value have meaning. In the future, more
		the bits will be added. Drag sources and drop targets sho
		these values appropriately before comparing. They shoul
		drop effect value against, say, DROPEFFECT_COPY by
		if DropEffect = DROPEFFECT_COPY then
		Instead, the application should always mask for the value or
		as using one of the following techniques:
		if bitwiseAnd(DropEffect, DROPEFFECT_COPY)
		FECT_COPY then
		This allows for the definition of new drop effects, while pr
		compatibility with existing code.
DROPEFFECT_LINK	4	One of the drop effect constants.
		Drag source should create a link to the original data.
DROPEFFECT_MOVE	2	One of the drop effect constants.
		Drag source should remove the data.
DROPEFFECT_NONE	0	One of the drop effect constants.
		Drop target cannot accept the data.
DROPEFFECT_SCROLL	&h80000000	One of the drop effect constants.
		Scrolling is about to start or is currently occurring in the ta
		used in addition to the other values.
MK_CONTROL	8	One of the key state constants.
		The CTRL key is down.
MK_LBUTTON	1	One of the key state constants.
		The left mouse button is down.
MK_MBUTTON	&h10	One of the key state constants.
		The middle mouse button is down.
MK_RBUTTON	2	One of the key state constants.
		The right mouse button is down.
MK_SHIFT	4	One of the key state constants.
		The SHIFT key is down.
MK_XBUTTON1	&h20	One of the key state constants.
		The first X button is down.
MK_XBUTTON2	&h40	One of the key state constants.
		The second X button is down.
S_FALSE	1	One of the OLE error codes.
S_OK	0	One of the OLE error codes.

12.3   class WindowsDropTargetMBS

12.3.1   class WindowsDropTargetMBS

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WindowsDropTargetMBS interface is one of the interfaces you implement to provide drag-and-drop operations in your application.
Notes: It contains methods used in any application that can be a target for data during a drag-and-drop operation. A drop-target application is responsible for:

- Determining the effect of the drop on the target application.

- Incorporating any valid dropped data when the drop occurs.

- Communicating target feedback to the source so the source application can provide appropriate visual feedback such as setting the cursor.

- Implementing drag scrolling.

- Registering and revoking its application windows as drop targets.

The WindowsDropTargetMBS class contains methods that handle all these responsibilities except registering and revoking the application window as a drop target, for which you must call the AttachToWindow functions.

When To Implement

Implement the WindowsDropTargetMBS interface if you are developing an application that can act as a target for a drag-and-drop operation. The WindowsDropTargetMBS interface is associated with your application windows and is implemented on your window objects. Call the AttachToWindow function to register your window objects as drop targets.

When To Use

You do not call the methods of WindowsDropTargetMBS directly. The DoDragDrop function calls the WindowsDropTargetMBS methods during the drag-and-drop operation.

For example, DoDragDrop calls WindowsDropTargetMBS.DragEnter when it detects the mouse has moved over a window that is registered as a drag target. After the mouse has entered a drag-target window, DoDragDrop calls WindowsDropTargetMBS.DragOver as the mouse moves through the window and calls WindowsDropTargetMBS.DragLeave if the mouse leaves the target window or if the user cancels or completes the drag-and-drop operation. DoDragDrop calls WindowsDropTargetMBS.Drop if the drop finally occurs.

To see the ghost picture of the drag, please register a WindowsDragSourceMBS for the same window.

While the drag classes compile for Web Edition, they run on the server, so they have no effect on the client browser!
Blog Entries

- MBS Real Studio Plugins, version 11.2pr5

- MBS REALbasic Plugins, version 10.5pr5

12.3.2 Methods

12.3.3 AttachToControl(ctl as control, showDragImage as boolean = true) as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop only.

Function: Attached the drop target to the control.
Notes: This method unregistered any existing drop target on the control (including the one from Xojo).

Registers the specified control as one that can be the target of an OLE drag-and-drop operation and specifies the WindowsDropTargetMBS instance to use for drop operations.

ctl: The control that can be a target for an OLE drag-and-drop operation.
showDragImage: Whether we should support the drag image methods in the newer Windows versions.

This function returns S_OK on success. Other possible values include the following.

Return code	Description
DRAGDROP_E_INVALIDHWND	Invalid handle returned in the hwnd parameter.
DRAGDROP_E_ALREADYREGISTERED	The specified window has already been registered as a drop target.
E_OUTOFMEMORY	Insufficient memory for the operation.

If your application can accept dropped objects during OLE drag-and-drop operations, you must call the AttachToControl function. Do this whenever one of your application windows is available as a potential drop target, i.e., when the window appears unobscured on the screen.

AttachToControl must be called on the main thread of your application.

The AttachToControl function only registers one control at a time, so you must call it for each application control capable of accepting dropped objects. For each control, you need your own instance of the Windows-DropTargetMBS class.

As the mouse passes over unobscured portions of the target control during an OLE drag-and-drop operation, the DoDragDrop function calls the specified WindowsDropTargetMBS.DragOver method for the current control. When a drop operation actually occurs in a given control, the DoDragDrop function calls WindowsDropTargetMBS.Drop.
See also:

- 12.3.4 AttachToControl(ctl as DesktopControl, showDragImage as boolean = true) as integer   400

12.3.4   AttachToControl(ctl as DesktopControl, showDragImage as boolean = true) as integer

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

Function: Attached the drop target to the control.
Notes: This method unregistered any existing drop target on the control (including the one from Xojo).

Registers the specified control as one that can be the target of an OLE drag-and-drop operation and specifies the WindowsDropTargetMBS instance to use for drop operations.

ctl: The control that can be a target for an OLE drag-and-drop operation.
showDragImage: Whether we should support the drag image methods in the newer Windows versions.

This function returns S_OK on success. Other possible values include the following.

Return code	Description
DRAGDROP_E_INVALIDHWND	Invalid handle returned in the hwnd parameter.
DRAGDROP_E_ALREADYREGISTERED	The specified window has already been registered as a drop target.
E_OUTOFMEMORY	Insufficient memory for the operation.

If your application can accept dropped objects during OLE drag-and-drop operations, you must call the AttachToControl function. Do this whenever one of your application windows is available as a potential drop target, i.e., when the window appears unobscured on the screen.

AttachToControl must be called on the main thread of your application.

The AttachToControl function only registers one control at a time, so you must call it for each application control capable of accepting dropped objects. For each control, you need your own instance of the WindowsDropTargetMBS class.

As the mouse passes over unobscured portions of the target control during an OLE drag-and-drop operation, the DoDragDrop function calls the specified WindowsDropTargetMBS.DragOver method for the current control. When a drop operation actually occurs in a given control, the DoDragDrop function calls Windows-

DropTargetMBS.Drop.
See also:

12.3.5 AttachToWindow(win as DesktopWindow, showDragImage as boolean = true) as integer

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

Function: Attached the drop target to the window.
Notes: This method unregistered any existing drop target on the window (including the one from Xojo).

Registers the specified window as one that can be the target of an OLE drag-and-drop operation and specifies the WindowsDropTargetMBS instance to use for drop operations.

Win: The window that can be a target for an OLE drag-and-drop operation.
showDragImage: Whether we should support the drag image methods in the newer Windows versions.

This function returns S_OK on success. Other possible values include the following.

Return code	Description
DRAGDROP_E_INVALIDHWND	Invalid handle returned in the hwnd parameter.
DRAGDROP_E_ALREADYREGISTERED	The specified window has already been registered as a drop target.
E_OUTOFMEMORY	Insufficient memory for the operation.

If your application can accept dropped objects during OLE drag-and-drop operations, you must call the AttachToWindow function. Do this whenever one of your application windows is available as a potential drop target, i.e., when the window appears unobscured on the screen.

AttachToWindow must be called on the main thread of your application.

The AttachToWindow function only registers one window at a time, so you must call it for each application window capable of accepting dropped objects. For each window, you need your own instance of the WindowsDropTargetMBS class.

As the mouse passes over unobscured portions of the target window during an OLE drag-and-drop operation, the DoDragDrop function calls the specified WindowsDropTargetMBS.DragOver method for the current window. When a drop operation actually occurs in a given window, the DoDragDrop function calls WindowsDropTargetMBS.Drop.
See also:

12.3.6   AttachToWindow(win as window, showDragImage as boolean = true) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop only.

Function: Attached the drop target to the window.
Notes: This method unregistered any existing drop target on the window (including the one from Xojo).

Registers the specified window as one that can be the target of an OLE drag-and-drop operation and specifies the WindowsDropTargetMBS instance to use for drop operations.

Win: The window that can be a target for an OLE drag-and-drop operation.
showDragImage: Whether we should support the drag image methods in the newer Windows versions.

This function returns S_OK on success. Other possible values include the following.

Return code	Description
DRAGDROP_E_INVALIDHWND	Invalid handle returned in the hwnd parameter.
DRAGDROP_E_ALREADYREGISTERED	The specified window has already been registered as a drop target.
E_OUTOFMEMORY	Insufficient memory for the operation.

If your application can accept dropped objects during OLE drag-and-drop operations, you must call the AttachToWindow function. Do this whenever one of your application windows is available as a potential drop target, i.e., when the window appears unobscured on the screen.

AttachToWindow must be called on the main thread of your application.

The AttachToWindow function only registers one window at a time, so you must call it for each application window capable of accepting dropped objects. For each window, you need your own instance of the WindowsDropTargetMBS class.

As the mouse passes over unobscured portions of the target window during an OLE drag-and-drop operation, the DoDragDrop function calls the specified WindowsDropTargetMBS.DragOver method for the current window. When a drop operation actually occurs in a given window, the DoDragDrop function calls WindowsDropTargetMBS.Drop.
See also:

12.3.7 Properties

12.3.8 Handle as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

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

12.3.9 Helper as Integer

Plugin Version: 10.5, Platform: Windows, Targets: Desktop, Console & Web.

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

12.3.10 Events

12.3.11 DragEnter(dataObject as WinDataObjectMBS, keystate as Integer, x as Integer, y as Integer, byref effect as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Indicates whether a drop can be accepted, and, if so, the effect of the drop.
Notes: dataObject: This data object contains the data being transferred in the drag-and-drop operation. If the drop occurs, this data object will be incorporated into the target.

KeyState: The current state of the keyboard modifier keys on the keyboard. Possible values can be a combination of any of the flags MK_CONTROL, MK_SHIFT, MK_ALT, MK_BUTTON, MK_LBUTTON, MK_MBUTTON, and MK_RBUTTON.

x and y: A point containing the current cursor coordinates in screen coordinates.

effect: The value of the Effect parameter of the DoDragDrop function. On return, must contain one of the DROPEFFECT flags, which indicates what the result of the drop operation would be.

Return S_OK on success. Other possible values include the following.

You do not call DragEnter directly; instead the DoDragDrop function calls it to determine the effect of a drop the first time the user drags the mouse into the registered window of a drop target.

Return code	Description
E_UNEXPECTED	An unexpected error has occurred.
E_INVALIDARG	The Effect parameter is NULL on input.
E_OUTOFMEMORY	There was insufficient memory available for this operation.

To implement DragEnter, you must determine whether the target can use the data in the source data object by checking three things:

- The format and medium specified by the data object

- The input value of Effect

- The state of the modifier keys

To check the format and medium, use the WinDataObjectMBS object.

On entry to WindowsDropTargetMBS.DragEnter, the Effect parameter is set to the effects given to the OkEffect parameter of the DoDragDrop function. The WindowsDropTargetMBS.DragEnter method must choose one of these effects or disable the drop.

The following modifier keys affect the result of the drop.

Key Combination	User-Visible Feedback	Drop Effect
CTRL + SHIFT	=	DROPEFFECT_LINK
CTRL	+	DROPEFFECT_COPY
No keys or SHIFT	None	DROPEFFECT_MOVE

On return, the method must write the effect, one of the DROPEFFECT flags, to the Effect parameter. DoDragDrop then takes this parameter and writes it to its Effect parameter. You communicate the effect of the drop back to the source through DoDragDrop in the Effect parameter. The DoDragDrop function then calls WindowsDragSourceMBS.GiveFeedback so that the source application can display the appropriate visual feedback to the user through the target window.

12.3.12   DragLeave as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Removes target feedback and releases the data object.
Notes: Return S_OK on success. Other possible values include the following.

Return code	Description
E_OUTOFMEMORY	There is insufficient memory available for this operation.

You do not call this method directly. The DoDragDrop function calls this method in either of the following cases:

- When the user drags the cursor out of a given target window.

- When the user cancels the current drag-and-drop operation.

To implement WindowsDropTargetMBS.DragLeave, you must remove any target feedback that is currently displayed. You must also release any references you hold to the data transfer object.

12.3.13 DragOver(keystate as Integer, x as Integer, y as Integer, byref effect as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Provides target feedback to the user and communicates the drop's effect to the DoDragDrop function so it can communicate the effect of the drop back to the source.
Notes: KeyState: The current state of the keyboard modifier keys on the keyboard. Valid values can be a combination of any of the flags MK_CONTROL, MK_SHIFT, MK_ALT, MK_BUTTON, MK_LBUTTON, MK_MBUTTON, and MK_RBUTTON.
x and y: The point containing the current cursor coordinates in screen coordinates.
Effect: On input, pointer to the value of the Effect parameter of the DoDragDrop function. On return, must contain one of the DROPEFFECT flags, which indicates what the result of the drop operation would be.

Return S_OK on success. Other possible values include the following.

Return code	Description
E_UNEXPECTED	An unexpected error has occurred.
E_INVALIDARG	The Effect value is not valid.
E_OUTOFMEMORY	There was insufficient memory available for this operation.

You do not call DragOver directly. The DoDragDrop function calls this method each time the user moves the mouse across a given target window. DoDragDrop exits the loop if the drag-and-drop operation is canceled, if the user drags the mouse out of the target window, or if the drop is completed.

In implementing WindowsDropTargetMBS.DragOver, you must provide features similar to those in WindowsDropTargetMBS.DragEnter. You must determine the effect of dropping the data on the target by

examining the FORMATETC defining the data object's formats and medium, along with the state of the modifier keys. The mouse position may also play a role in determining the effect of a drop. The following modifier keys affect the result of the drop.

Key Combination	User-Visible Feedback	Drop Effect
CTRL + SHIFT	=	DROPEFFECT_LINK
CTRL	+	DROPEFFECT_COPY
No keys or SHIFT	None	DROPEFFECT_MOVE

You communicate the effect of the drop back to the source through DoDragDrop in Effect. The DoDrag-Drop function then calls WindowsDragSourceMBS.GiveFeedback so the source application can display the appropriate visual feedback to the user.

On entry to WindowsDropTargetMBS.DragOver, the Effect parameter must be set to the allowed effects passed to the OkEffect parameter of the DoDragDrop function. The WindowsDropTargetMBS.DragOver method must be able to choose one of these effects or disable the drop.

Upon return, Effect is set to one of the DROPEFFECT flags. This value is then passed to the Effect parameter of DoDragDrop. Reasonable values are DROPEFFECT_COPY to copy the dragged data to the target, DROPEFFECT_LINK to create a link to the source data, or DROPEFFECT_MOVE to allow the dragged data to be permanently moved from the source application to the target.

You may also wish to provide appropriate visual feedback in the target window. There may be some target feedback already displayed from a previous call to WindowsDropTargetMBS.DragOver or from the initial WindowsDropTargetMBS.DragEnter. If this feedback is no longer appropriate, you should remove it.

For efficiency reasons, a data object is not passed in WindowsDropTargetMBS.DragOver. The data object passed in the most recent call to WindowsDropTargetMBS.DragEnter is available and can be used.

When WindowsDropTargetMBS.DragOver has completed its operation, the DoDragDrop function calls WindowsDragSourceMBS.GiveFeedback so the source application can display the appropriate visual feedback to the user.

Notes to Implementers

This function is called frequently during the DoDragDrop loop so it makes sense to optimize your implementation of the DragOver method as much as possible.

12.3.14 Drop(dataObject as WinDataObjectMBS, keystate as Integer, x as Integer, y as Integer, byref effect as Integer) as Integer

Plugin Version: 10.5, Platform: Windows, Targets: .

Function: Incorporates the source data into the target window, removes target feedback, and releases the data object.
Notes: dataObject: The data object being transferred in the drag-and-drop operation.
KeyState: The current state of the keyboard modifier keys on the keyboard. Possible values can be a combination of any of the flags MK_CONTROL, MK_SHIFT, MK_ALT, MK_BUTTON, MK_LBUTTON, MK_MBUTTON, and MK_RBUTTON.
x and y: The point containing the current cursor coordinates in screen coordinates.
Effect: On input, the value of the Effect parameter of the DoDragDrop function. On return, must contain one of the DROPEFFECT flags, which indicates what the result of the drop operation would be.

Return S_OK on success. Other possible values include the following.

Return code	Description
E_UNEXPECTED	An unexpected error has occurred.
E_INVALIDARG	The pdwEffect parameter is not valid.
E_OUTOFMEMORY	There is insufficient memory available for this operation.

You do not call this method directly. The DoDragDrop function calls this method when the user completes the drag-and-drop operation.

In implementing Drop, you must incorporate the data object into the target. Use the formats available in WinDataObjectMBS, available through dataObject, along with the current state of the modifier keys to determine how the data is to be incorporated, such as linking or embedding.

In addition to incorporating the data, you must also clean up as you do in the WindowsDropTargetMBS.DragLeave method:

Remove any target feedback that is currently displayed.
Release any references to the data object.
You also pass the effect of this operation back to the source application through DoDragDrop, so the source application can clean up after the drag-and-drop operation is complete:

Remove any source feedback that is being displayed.
Make any necessary changes to the data, such as removing the data if the operation was a move.

12.3.15   Constants

Constants

Constant	Value	Description
DROPEFFECT_COPY	1	One of the drop effect constants.
		Drop results in a copy. The original data is untouched by the drag source. Your application should always mask drop effect constants to ensure compatibility with future implementations. Presently, only some of the bit positions in a drop effect value have meaning. In the future, more interpretations for the bits will be added. Drag sources and drop targets should carefully mask these values appropriately before comparing. They should never compare a drop effect value against, say, DROPEFFECT_COPY by doing the following: if DropEffect = DROPEFFECT_COPY then Instead, the application should always mask for the value or values being sought as using one of the following techniques: if bitwiseAnd(DropEffect, DROPEFFECT_COPY) = DROPEFFECT_COPY then This allows for the definition of new drop effects, while preserving backward compatibility with existing code.
DROPEFFECT_LINK	4	One of the drop effect constants.
		Drag source should create a link to the original data.
DROPEFFECT_MOVE	2	One of the drop effect constants.
		Drag source should remove the data.
DROPEFFECT_NONE	0	One of the drop effect constants.
		Drop target cannot accept the data.
DROPEFFECT_SCROLL	&h80000000	One of the drop effect constants.
		Scrolling is about to start or is currently occurring in the target. This value used in addition to the other values.
E_INVALIDARG	&h80070057	One of the OLE error codes.
		An invalid argument was passed.
E_OUTOFMEMORY	&h80000002	One of the OLE error codes.
		There was insufficient memory available for this operation.
E_UNEXPECTED	&h8000FFFF	One of the OLE error codes.
		An unexpected error has occurred.
MK_CONTROL	8	One of the key state constants.
		The CTRL key is down.
MK_LBUTTON	1	One of the key state constants.
		The left mouse button is down.
MK_MBUTTON	&h10	One of the key state constants.
		The middle mouse button is down.
MK_RBUTTON	2	One of the key state constants.
		The right mouse button is down.
MK_SHIFT	4	One of the key state constants.
		The SHIFT key is down.
MK_XBUTTON1	&h20	One of the key state constants.
		The first X button is down.
MK_XBUTTON2	&h40	One of the key state constants.
		The second X button is down.
S_FALSE	1	One of the OLE error codes.
S_OK	0	One of the OLE error codes.

12.4   class WindowsFileDescriptorMBS

12.4.1   class WindowsFileDescriptorMBS

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a file description.s
Blog Entries

- MBS Real Studio Plugins, version 11.2pr4

12.4.2   Properties

12.4.3   ClassID as String

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file type identifier.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_CLSID) <>0 then
MsgBox d.ClassID
end if

Notes: Only valid if FD_CLSID is set in the flags.
(Read and Write property)

12.4.4   CreationTime as Double

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time of file creation.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_CREATETIME) <>0 then
dim da as new date
da.TotalSeconds = d.CreationTime
MsgBox da.LongDate
end if

Notes: Only valid if FD_CREATETIME is set in the flags.
(Read and Write property)

12.4.5 FileAttributes as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: File attribute flags.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_ATTRIBUTES) <>0 then
if BitwiseAnd(d.FileAttributes, d.FILE_ATTRIBUTE_TEMPORARY) = d.FILE_ATTRIBUTE_TEM-
PORARY then
MsgBox "temp file"
else
MsgBox "no temp file"
end if
end if

Notes: This will be a combination of the FILE_ATTRIBUTE_* constants.
Only valid if FD_ATTRIBUTES is set in the flags.
(Read and Write property)

12.4.6 FileName as String

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that contains the name of the file.
Example:

dim d as WindowsFileDescriptorMBS // your file description

MsgBox d.FileName

Notes: (Read and Write property)

12.4.7   FileSize as Int64

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file size, in bytes.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_FILESIZE) <>0 then
MsgBox str(d.FileSize)
end if

Notes: Only valid if FD_FILESIZE is set in flags.
(Read and Write property)

12.4.8   Flags as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: An array of flags that indicate which of the other structure members contain valid data.
Notes: A combination of the FD_* constants.
(Read and Write property)

12.4.9   IconHeight as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the file icon.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_SIZEPOINT) <>0 then
MsgBox "file object at "+stR(d.Pointx)+"/"+str(d.Pointy)+" with size "+str(d.IconWidth)+"/"+str(d.Icon-
Height)
end if

Notes: Only valid if FD_SIZEPOINT is set in the flags.
(Read and Write property)

12.4.10 IconWidth as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the file icon.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_SIZEPOINT) <>0 then
MsgBox "file object at "+stR(d.Pointx)+"/"+str(d.Pointy)+" with size "+str(d.IconWidth)+"/"+str(d.Icon-Height)
end if

Notes: Only valid if FD_SIZEPOINT is set in the flags.
(Read and Write property)

12.4.11 Index as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The index of the file descriptor.
Notes: Use this entry for GetFileContent call.
(Read and Write property)

12.4.12 LastAccessTime as Double

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time that the file was last accessed.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_ACCESSTIME) <>0 then
dim da as new date
da.TotalSeconds = d.LastAccessTime
MsgBox da.LongDate
end if

Notes: Only valid if FD_ACCESSTIME is set in the flags.
(Read and Write property)

12.4.13   LastWriteTime as Double

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time of the last write operation.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_WRITESTIME) <>0 then
dim da as new date
da.TotalSeconds = d.LastWriteTime
MsgBox da.LongDate
end if

Notes: Only valid if FD_WRITESTIME is set in the flags.
(Read and Write property)

12.4.14   PointX as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The x screen coordinate of the file object.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_SIZEPOINT) <>0 then
MsgBox "file object at "+stR(d.Pointx)+"/"+str(d.Pointy)+" with size "+str(d.IconWidth)+"/"+str(d.Icon-
Height)
end if

Notes: Only valid if FD_SIZEPOINT is set in the flags.
(Read and Write property)

12.4.15   PointY as Integer

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The y screen coordinate of the file object.
Example:

dim d as WindowsFileDescriptorMBS // your file description

if BitwiseAnd(d.Flags, d.FD_SIZEPOINT) <>0 then
MsgBox "file object at "+stR(d.Pointx)+"/"+str(d.Pointy)+" with size "+str(d.IconWidth)+"/"+str(d.Icon-
Height)
end if

Notes: Only valid if FD_SIZEPOINT is set in the flags.
(Read and Write property)

12.4.16 Constants

Flag constants

Constant	Value	Description
FD_ACCESSTIME	&h0010	The LastAccessTime member is valid.
FD_ATTRIBUTES	4	The FileAttributes member is valid.
FD_CLSID	1	The ClassID member is valid.
FD_CREATETIME	8	The CreationTime member is valid.
FD_FILESIZE	&h0040	Whether the FileSize member is valid.
FD_LINKUI	&h8000	Treat the operation as a shortcut.
FD_PROGRESSUI	&h4000	A progress indicator is shown with drag-and-drop operations.
FD_SIZEPOINT	2	The Icon* and point* members are valid.
FD_WRITESTIME	&h0020	Whether the LastWriteTime property is valid.

File attribute constants

Constant	Value	Description
FILE_ATTRIBUTE_ARCHIVE	&h00000020	A file or directory that is an archive file or directory. Appl use this attribute to mark files for backup or removal .
FILE_ATTRIBUTE_ATOMIC_WRITE	&h00000200	
FILE_ATTRIBUTE_COMPRESSED	&h00000800	A file or directory that is compressed. For a file, all of the compressed. For a directory, compression is the default for n and subdirectories.
FILE_ATTRIBUTE_DIRECTORY	&h00000010	The handle that identifies a directory.
FILE_ATTRIBUTE_HIDDEN	&h00000002	The file or directory is hidden. It is not included in an o listing.
FILE_ATTRIBUTE_NORMAL	&h00000080	A file that does not have other attributes set. This attribute i used alone.
FILE_ATTRIBUTE_OFFLINE	&h00001000	The data of a file is not available immediately. This attribute file data is physically moved to offline storage. This attribute i Storage, which is the hierarchical storage management softw should not arbitrarily change this attribute.
FILE_ATTRIBUTE_READONLY	&h00000001	A file that is read-only. Applications can read the file, but c or delete it.
FILE_ATTRIBUTE_SYSTEM	&h00000004	A file or directory that the operating system uses a part of, o
FILE_ATTRIBUTE_TEMPORARY	&h00000100	A file that is being used for temporary storage. File systems a back to mass storage if sufficient cache memory is available, an application deletes a temporary file after the handle is scenario, the system can entirely avoid writing the data. Ot is written after the handle is closed.
FILE_ATTRIBUTE_XACTION_WRITE	&h00000400	

Chapter 13

Files

13.1 Globals

13.1.1 WindowsEjectVolumeMBS(driveLetter as string, byref status as Integer) as boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Unmounts and ejects the given drive.
Example:

```
// editfield1 has the drive letter

dim status as Integer
dim b as Boolean = WindowsEjectVolumeMBS(EditField1.text, status)

if b then
Select case status
case 1
MsgBox "Media in Drive "+EditField1.text+" has been ejected safely."
case 2
MsgBox "Media in Drive "+EditField1.text+" can be safely removed."
else
MsgBox "Failed?"
end Select
else
MsgBox "Failed."
end if
```

Notes: Status is set to 1 after an eject or 2 after an unmount without eject.

Returns true on success and false on failure.
Blog Entries

- MBS REALbasic Plugins, version 10.6pr4

13.1.2   GetDriveTypeMBS(path as string) as Integer

Plugin Version: 12.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Determines whether a disk drive is a removable, fixed, CD-ROM, RAM disk, or network drive.
Example:

if GetDriveTypeMBS("E:") = 4 then
msgbox "network drive"
end if

Notes: Path: The root directory for the drive.

Possible values:

0	Unknown	The drive type cannot be determined.
1	Error	The root path is invalid; for example, there is no volume mounted at the specified path.
2	Removable	The drive has removable media; for example, a floppy drive, thumb drive, or flash card reader.
3	Fixed	The drive has fixed media; for example, a hard disk drive or flash drive.
4	Remote	The drive is a remote (network) drive.
5	CD-ROM	The drive is a CD-ROM drive.
6	RAM Disk	The drive is a RAM disk.

See also IsOnRemoteVolumeMBS, IsEjectableVolume or DarwinMediaClassMBS for macOS.
Blog Entries

- MBS Real Studio Plugins, version 12.3pr2

13.2   class FolderItem

13.2.1   class FolderItem

Platforms: macOS, Linux, Windows, Targets: All.

Function: One of Xojo's base classes.
Notes: Handles access to files.

13.2.2 Methods

13.2.3 LongPathMBS as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the long path for the file.
Example:

dim f as folderitem = specialfolder.desktop.child("test.file")
msgbox f.longpathMBS

Notes: Should be the same as f.NativePath
Works with Windows NT 4 or newer.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr8

Xojo Developer Magazine

- 12.2, page 28: Smart Reporting, Implementing a custom reporting system by Trisha Duke

13.2.4 ShortPathMBS as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the short path for the file.
Example:

dim f as folderitem = specialfolder.desktop.child("test.file")
msgbox f.ShortPathMBS

Notes: In contrast to long path this is the short 8.3 path for Windows.
You need this for the WindowsMCI object.
Works with Windows NT 4 or newer.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr8

13.2.5   WinThumbnailMBS(preferredSize as Integer = 512) as picture

Plugin Version: 14.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries a thumbnail for an item.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
Backdrop = f.WinThumbnailMBS

Notes: preferredSize is the size you'd like to have. Resulting image can be smaller or bigger.
Returns nil on any error.
If user disabled thumbnails for explorer, the shell also provides none for us, just icons.
Requires Windows Vista or newer.
Blog Entries

- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.3

- MBS Xojo / Real Studio Plugins, version 14.3pr1

13.3 class WindowsDiskChangeMBS

13.3.1 class WindowsDiskChangeMBS

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to get events for media or devices being added/inserted or removed.
Blog Entries

- MonkeyBread Software Releases the MBS Real Studio plug-ins in version 13.1
- MBS Real Studio Plugins, version 13.1pr10

13.3.2 Methods

13.3.3 Constructor

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

13.3.4 Properties

13.3.5 Valid as Boolean

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the constructor registered the events successfully.
Notes: (Read and Write property)

13.3.6 Events

13.3.7 DriveAdded(Path as string)

Plugin Version: 13.1, Platform: Windows, Targets: .

Function: The event to notify you about drive being added.

13.3.8 DriveRemoved(Path as string)

Plugin Version: 13.1, Platform: Windows, Targets: .

Function: The event to notify you about drive being removed.

13.3.9   MediaInserted(Path as string)

Plugin Version: 13.1, Platform: Windows, Targets: .

Function: The event to notify you about media being added.

13.3.10   MediaRemoved(Path as string)

Plugin Version: 13.1, Platform: Windows, Targets: .

Function: The event to notify you about media being removed.
Notes: You may get the event multiple times.

13.4 class WindowsDriveNotificationMBS

13.4.1 class WindowsDriveNotificationMBS

Plugin Version: 15.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to detect devices being mounted/unmounted in Windows.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.0

- MBS Xojo / Real Studio Plugins, version 14.5pr1

13.4.2 Events

13.4.3 DeviceArrival(Path as string)

Plugin Version: 15.0, Platform: Windows, Targets: .

Function: A device arrived.
Notes: Path my be path to USB device or path to mount point. For USB sticks you get both.

13.4.4 DeviceRemoved(Path as string)

Plugin Version: 15.0, Platform: Windows, Targets: .

Function: A device was removed.
Notes: Path my be path to USB device or path to mount point. For USB sticks you get both.

13.5 module WindowsJunctionMBS

13.5.1 module WindowsJunctionMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: This module implements various functions for links.
Notes: Symbol links (link files), Hard links (several directory entries for one file on disc) and junctions (show content of one folder inside another folder).

Some operations needs Administrator permissions.
Check lasterror in case of trouble.
Error 1314 for example is permissions problem.

See also WindowsShortCutMBS and WindowsInternetShortCutMBS classes.
For Mac see also CFBookmarkMBS and MacAliasMBS classes.
Blog Entries

- MBS Xojo Plugins, version 19.1pr7

- MBS Real Studio Plugins, version 11.3pr14

13.5.2 Methods

13.5.3 CreateHardLink(NewFile as folderitem, TargetFile as folderitem) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Establishes a hard link between an existing file and a new file.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.exe")
dim nfile as FolderItem = SpecialFolder.Desktop.Child("hello.exe")

if WindowsJunctionMBS.CreateHardLink(nfile, file) then
MsgBox "OK"
end if

Notes: This function is only supported on the NTFS file system, and only for files, not directories.

NewFile: The name of the new file. Function fails if NewFile exists already. This parameter cannot specify the name of a directory.
TargetFile: The name of the existing file. This parameter cannot specify the name of a directory.

Returns true on success. Lasterror is set.

The maximum number of hard links that can be created with this function is 1023 per file. If more than 1023 links are created for a file, an error results.

Any directory entry for a file that is created with CreateFile or CreateHardLink is a hard link to an associated file. An additional hard link that is created with the CreateHardLink function allows you to have multiple directory entries for a file, that is, multiple hard links to the same file, which can be different names in the same directory, or the same or different names in different directories. However, all hard links to a file must be on the same volume.

Because hard links are only directory entries for a file, many changes to that file are instantly visible to applications that access it through the hard links that reference it. However, the directory entry size and attribute information is updated only for the link through which the change was made.

The security descriptor belongs to the file to which a hard link points. The link itself is only a directory entry, and does not have a security descriptor. Therefore, when you change the security descriptor of a hard link, you a change the security descriptor of the underlying file, and all hard links that point to the file allow the newly specified access. You cannot give a file different security descriptors on a per-hard-link basis.

Use DeleteFile to delete hard links (folderitem.delete in Xojo). You can delete them in any order regardless of the order in which they are created.

Flags, attributes, access, and sharing that are specified in CreateFile operate on a per-file basis. That is, if you open a file that does not allow sharing, another application cannot share the file by creating a new hard link to the file.

When you create a hard link on the NTFS file system, the file attribute information in the directory entry is refreshed only when the file is opened, or when GetFileInformationByHandle is called with the handle of a specific file.

Symbolic link behavior—If the path points to a symbolic link, the function creates a hard link to the target.

Lasterror is set. Returns true on success.

13.5.4 CreateJunction(JunctionDir as folderitem, TargetDir as folderitem) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new junction.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test")
dim nfile as FolderItem = SpecialFolder.System

if WindowsJunctionMBS.CreateJunction(file, nfile) then
MsgBox "OK"
end if

Notes: Lasterror is set. Returns true on success.

13.5.5   CreateSymbolicLink(NewFile as folderitem, TargetFile as folderitem) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a symbolic link.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.lnk")
dim nfile as FolderItem = SpecialFolder.Desktop.Child("test.txt")

if WindowsJunctionMBS.CreateSymbolicLink(file,nfile) then
MsgBox "Ok"
end if

Notes: NewFile: The symbolic link to be created.
TargetFile: The name of the target for the symbolic link to be created. If TargetFile has a device name associated with it, the link is treated as an absolute link; otherwise, the link is treated as a relative link.
TargetIsDirectory: Indicates whether the link target, TargetFile, is a directory. (Pass true for a directory).

Returns true on success. False on failure. Lasterror is set.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

Symbolic links can either be absolute or relative links. Absolute links are links that specify each portion of the path name; relative links are determined relative to where relative–link specifiers are in a specified path. Relative links are specified using the following conventions:

- Dot (. and ..) conventions—for example, "..\" resolves the path relative to the parent directory.

- Names with no slashes (\)—for example, "tmp" resolves the path relative to the current directory.

- Root relative—for example, "\Windows\System32" resolves to "current drive:\Windows\System32".

- Current working directory–relative—for example, if the current working directory is C:\Windows\System32, "C:File.txt" resolves to "C:\Windows\System32\File.txt".

If you specify a current working directory–relative link, it is created as an absolute link, due to the way the

current working directory is processed based on the user and the thread.

This function can fail due to missing permissions.
See also:

13.5.6 CreateSymbolicLink(NewFile as folderitem, TargetFile as string, TargetIsDirectory as Boolean) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a symbolic link.
Notes: NewFile: The symbolic link to be created.
TargetFile: The name of the target for the symbolic link to be created. If TargetFile has a device name associated with it, the link is treated as an absolute link; otherwise, the link is treated as a relative link.
TargetIsDirectory: Indicates whether the link target, TargetFile, is a directory. (Pass true for a directory).

Returns true on success. False on failure. Lasterror is set.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

Symbolic links can either be absolute or relative links. Absolute links are links that specify each portion of the path name; relative links are determined relative to where relative–link specifiers are in a specified path. Relative links are specified using the following conventions:

- Dot (. and ..) conventions—for example, "..\" resolves the path relative to the parent directory.

- Names with no slashes (\)—for example, "tmp" resolves the path relative to the current directory.

- Root relative—for example, "\Windows\System32" resolves to "current drive:\Windows\System32".

- Current working directory–relative—for example, if the current working directory is C:\Windows\System32, "C:File.txt" resolves to "C:\Windows\System32\File.txt".

If you specify a current working directory–relative link, it is created as an absolute link, due to the way the current working directory is processed based on the user and the thread.

This function can fail due to missing permissions.
See also:

13.5.7   DeleteJunction(JunctionDir as folderitem) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes the junction.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test")

if WindowsJunctionMBS.DeleteJunction(file) then
MsgBox "Ok"
end if

Notes: Folder is not deleted.

13.5.8   GetJunctionTarget(JunctionDir as folderitem) as string

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether a directory is a directory junction.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test")

msgbox WindowsJunctionMBS.GetJunctionTarget(file)

Notes: Lasterror is set. Returns path of junction target.

13.5.9   HardLinksForFile(path as string) as string()

Plugin Version: 19.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Finds all the hard links to the specified file.
Example:

dim path as string = "C:\Users\Christian\Desktop\test.xojo_binary_project"

dim oldfile as FolderItem = GetFolderItem(path, FolderItem.PathTypeNative)
dim newfile as FolderItem = GetFolderItem(path+".copy", FolderItem.PathTypeNative)

call WindowsJunctionMBS.CreateHardLink(newfile, oldfile)

dim paths() as string = WindowsJunctionMBS.HardLinksForFile(path)

MsgBox Join(paths, EndOfLine)

Notes: Path: The name of the file.

Array has only one entry if file has no other hard links.

13.5.10 IsDirectoryJunction(JunctionDir as folderitem) as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether a directory is a directory junction.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test")

if WindowsJunctionMBS.IsDirectoryJunction(file) then
MsgBox "Is junction."
else
MsgBox "Is no junction."
end if

Notes: Returns true if this file is a junction.
Lasterror is set.

13.5.11 Lasterror as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the last error code.
Notes: Value is 0 for no error
-1 for some parameter error or not implemented inside plugin.
other values are windows error codes.

13.6   class WindowsVolumeInformationMBS

13.6.1   class WindowsVolumeInformationMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Detailed information on Windows volumes.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.2pr1

13.6.2   Methods

13.6.3   Constructor

Plugin Version: 5.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A dummy constructor used only for automatic plugin testing.
Notes: Uses "C:" as path and does the same as the other constructor.
See also:

13.6.4   Constructor(path as string)

Platform: Windows, Targets: Desktop, Console & Web.

Function: Fills the class with information about the given volume.
Example:

Function Window1.ToString(v as Variant) As string
return v
End Function

Sub Window1.Open
dim w as WindowsVolumeInformationMBS

w=new WindowsVolumeInformationMBS("C:\")

pname.text=w.name
pfsName.text=w.FileSystemName
pmaxLen.text=tostring(w.maxNameLength)
pcasePres.text=tostring(w.CaseIsPreserved)
pcasetiv.text=tostring(w.caseSensitive)
punicode.text=tostring(w.SupportsUnicodeFilenames)
pcompressed.text=tostring(w.IsCompressedVolume)

pfileCompress.text=tostring(w.SupportsFileCompression)
pfileEncrypt.text=tostring(w.SupportsFileEncryption)
pserial.text=hex(w.serial)
End Sub

Notes: Path must something like "C:\" or "\\MyServer\MyShare\".
See also:

13.6.5 Properties

13.6.6 CaseIsPreserved as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file system preserves the case of filenames when it places a name on disk.
Notes: (Read only property)

13.6.7 CaseSensitive as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file system supports case-sensitive filenames.
Notes: (Read only property)

13.6.8 FileSystemName as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the filesystem.
Notes: For example "FAT" or "NTFS".
(Read only property)

13.6.9 IsCompressedVolume as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The specified volume is a compressed volume; for example a DoubleSpace volume.
Notes: A volume is only SupportsFileCompression or IsCompressedVolume, but never both together.

(Read only property)

13.6.10 MaxNameLength as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum length of a file name on this volume.
Notes: Should be 255 on normal Windows disks.
(Read only property)

13.6.11 Name as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The volume name or label if the volume is named.
Notes: (Read only property)

13.6.12 Path as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The path used in the constructor.
Notes: (Read only property)

13.6.13 Serial as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The volume serial number.
Example:

dim w as WindowsVolumeInformationMBS

w=new WindowsVolumeInformationMBS("C:\")
MsgBox hex(w.Serial)

Notes: This serial is set to a random name on formatting and can be changed later using some special tools.
(Read only property)

13.6.14 SupportsFileCompression as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file system supports file-based compression.
Notes: A volume is only SupportsFileCompression or IsCompressedVolume, but never both together.
(Read only property)

13.6.15 SupportsFileEncryption as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file system supports the Encrypted File System (EPS).
Notes: (Read only property)

13.6.16 SupportsUnicodeFilenames as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file system supports Unicode in filenames as they appear on disk.
Notes: (Read only property)

13.6.17 Valid as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Was the constructor successfull?
Notes: May fail on bad file paths.
(Read only property)

Chapter 14

Folder Change Watching

14.1   class WindowsDirectoryChangeMBS

14.1.1   class WindowsDirectoryChangeMBS

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for directory change information.
Notes: For Mac, you can use FSEventsMBS class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.5pr4

14.1.2   Properties

14.1.3   Action as Integer

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The type of change that has occurred.
Notes: See kAction constants.
(Read and Write property)

14.1.4   Filename as String

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file path.
Notes: For speed reasons we give you string, not folderitem.

(Read and Write property)

14.1.5   Constants

Type Change Constants

Constant	Value	Description
kActionAdded	1	The file was added to the directory.
kActionModified	3	The file was modified. This can be a change in the time stamp or attributes.
kActionRemoved	2	The file was removed from the directory.
kActionRenamedNewName	5	The file was renamed and this is the new name.
kActionRenamedOldName	4	The file was renamed and this is the old name.

14.2 class WindowsDirectoryWatcherMBS

14.2.1 class WindowsDirectoryWatcherMBS

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for watching folders on Windows for changes.
Notes: Create an object, add a few directories you want to watch and wait for changes being reported by NextChange method.
Blog Entries

- MBS Releases the MBS Real Studio plug-ins in version 12.0

- MBS Real Studio Plugins, version 11.4pr3

Xojo Developer Magazine

- 10.3, page 9: News

14.2.2 Methods

14.2.3 AddDirectory(path as folderitem, Recursive as boolean, Flags as Integer) as Boolean

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Schedules a directory for watching.
Notes: path: Path to folder to watch.
Recursive: Whether to watch subdirectories.
Flags: What to watch for. Please combine kNotifyChange* constants with bitwiseOr.

Returns false on error and true on success.
You can add several directories to a watcher.
See also:

- 14.2.4 AddDirectory(path as string, Recursive as boolean, Flags as Integer) as Boolean   437

14.2.4 AddDirectory(path as string, Recursive as boolean, Flags as Integer) as Boolean

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Schedules a directory for watching.
Notes: path: Path to folder to watch.
Recursive: Whether to watch subdirectories.

Flags: What to watch for. Please combine kNotifyChange* constants with bitwiseOr.

Returns false on error and true on success.
You can add several directories to a watcher.
See also:

14.2.5   Constructor

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

14.2.6   NextChange as WindowsDirectoryChangeMBS

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries changes.
Notes: Call this method in a loop in a timer until it returns nil.
This way you can get all the changes.

14.2.7   Constants

Flags

Constant	Value	Description
kNotifyChangeAttributes	4	Any attribute change in the watched directory or subtree causes a change notification wait operation to return.
kNotifyChangeCreation	64	Any change to the creation time of files in the watched directory or subtree causes a change notification wait operation to return.
kNotifyChangeDirName	2	Any directory-name change in the watched directory or subtree causes a change notification wait operation to return. Changes include creating or deleting a directory.
kNotifyChangeFilename	1	Any file name change in the watched directory or subtree causes a change notification wait operation to return. Changes include renaming, creating, or deleting a file.
kNotifyChangeLastAccess	32	Any change to the last access time of files in the watched directory or subtree causes a change notification wait operation to return.
		Be careful: This flag is very performance consuming as you get a lot of events!
kNotifyChangeLastWrite	16	Any change to the last write-time of files in the watched directory or subtree causes a change notification wait operation to return. The operating system detects a change to the last write-time only when the file is written to the disk. For operating systems that use extensive caching, detection occurs only when the cache is sufficiently flushed.
kNotifyChangeSecurity	256	Any security-descriptor change in the watched directory or subtree causes a change notification wait operation to return.
kNotifyChangeSize	8	Any file-size change in the watched directory or subtree causes a change notification wait operation to return. The operating system detects a change in file size only when the file is written to the disk. For operating systems that use extensive caching, detection occurs only when the cache is sufficiently flushed.

14.3　class WindowsFolderChangeMBS

14.3.1　class WindowsFolderChangeMBS

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to get notifications for folder changes on Windows.
Notes: On Mac OS X 10.5 you can use the FSEventsMBS class and on older Mac OS X the class Folder-ChangedNotificationMBS.

See also newer WindowsDirectoryWatcherMBS class.
Blog Entries

- MBS REALbasic plug-in 9.6

14.3.2　Methods

14.3.3　Constructor(path as folderitem, subtree as boolean, FilterFlags as Integer)

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor to start the notification service.
Notes: path: The full path of the directory to be watched.
subtree: If this parameter is true, the function monitors the directory tree rooted at the specified directory; if it is false, it monitors only the specified directory.
FilterFlags: The filter conditions that satisfy a change notification wait. This parameter can be one or more of the constants in this class.

Requires Windows 2000.

14.3.4　Properties

14.3.5　ChangeCount as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of calls to the Changed event.
Notes: (Read and Write property)

14.3.6 Handle as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal handle of the notification.
Notes: (Read and Write property)

14.3.7 Events

14.3.8 Changed

Plugin Version: 8.5, Platform: Windows, Targets: .

Function: Something has changed in the given folder.

14.3.9 Constants

Constants

Constant	Value	Description
ChangeAttribute	4	One of the flags for the constructor.
		Any attribute change in the watched directory or subtree causes a change notification wait operation to return.
ChangeDir	2	One of the flags for the constructor.
		Any directory-name change in the watched directory or subtree causes a change notification wait operation to return. Changes include creating or deleting a directory.
ChangeFile	1	One of the flags for the constructor.
		Any file name change in the watched directory or subtree causes a change notification wait operation to return. Changes include renaming, creating, or deleting a file name.
ChangeSecurity	256	One of the flags for the constructor.
		Any security-descriptor change in the watched directory or subtree causes a change notification wait operation to return.
ChangeSize	8	One of the flags for the constructor.
		Any file-size change in the watched directory or subtree causes a change notification wait operation to return. The operating system detects a change in file size only when the file is written to the disk. For operating systems that use extensive caching, detection occurs only when the cache is sufficiently flushed.
ChangeWrite	16	One of the flags for the constructor.
		Any change to the last write-time of files in the watched directory or subtree causes a change notification wait operation to return. The operating system detects a change to the last write-time only when the file is written to the disk. For operating systems that use extensive caching, detection occurs only when the cache is sufficiently flushed.

Chapter 15

Fonts

15.1   class WindowsFontDialogMBS

15.1.1   class WindowsFontDialogMBS

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The class for the standard font dialog for Windows.
Notes: Creates a Font dialog box that enables the user to choose attributes for a logical font. These attributes include a font family and associated font style, a point size, effects (underline, strikeout, and text color), and a script (or character set).

see also
https://msdn.microsoft.com/en-us/library/windows/desktop/ms646914(v=vs.85).aspx
Blog Entries

- MBS Xojo Plugins, version 21.6pr3

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.3

- MBS Xojo Plugins, version 17.3pr5

Videos

- Presentation from Munich conference about MBS Plugins.

- Presentation from London conference about MBS Plugins.

Xojo Developer Magazine

- 15.5, page 9: News

15.1.2   Methods

15.1.3   ChooseFont as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Creates a Font dialog box that enables the user to choose attributes for a logical font.
Notes: Returns true if user pressed OK bitton.
Lasterror is set.
Returns false als sets lasterror to zero for cancel.

15.1.4   CloseDialog

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Close the dialog.
Notes: Can be called from various events to close the dialog programmatically.

15.1.5   Query

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Queries current dialog values into properties.

15.1.6   Update

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Updates dialog with current properties.

15.1.7   Properties

15.1.8   Bold as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The bold value.
Notes: (Read and Write property)

15.1.9 CurrentFont as WindowsFontFamilyMBS

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Returns current font.
Notes: You may need to call Query to get current state from dialog.
This returns a copy, so changes to the font object are not going back.
(Read only property)

15.1.10 DialogHandle as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Internal window handle for dialog.
Notes: (Read and Write property)

15.1.11 Effects as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to show text effect controls.
Notes: Causes the dialog box to display the controls that allow the user to specify strikeout, underline, and text color options. If this flag is set, you can use the textColor member to specify the initial text color. You can use the StrikeThrough and Underline members to specify the initial settings of the strikeout and underline check boxes. ChooseFont can use these members to return the user's selections.
(Read and Write property)

15.1.12 FontName as String

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The font name of current font.
Notes: (Read and Write property)

15.1.13 FontType as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The type of the selected font when ChooseFont returns.
Notes: This member can be one or more of the following values:
FontTypeBold, FontTypeItalic, FontTypePrinter, FontTypeRegular, FontTypeScreen and FontTypeSimulator.

(Read and Write property)

15.1.14   ForceFontExist as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to only allow existing fonts.
Notes: ChooseFont should indicate an error condition if the user attempts to select a font or style that is not listed in the dialog box.
(Read and Write property)

15.1.15   Height as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The height, in logical units, of the font's character cell or character.
Notes: The character height value (also known as the em height) is the character cell height value minus the internal-leading value. The font mapper interprets the value specified in lfHeight in the following manner.

Value	Meaning
>0	The font mapper transforms this value into device units and matches it against the cell height of the available fonts.
0	The font mapper uses a default height value when it searches for a match.
<0	The font mapper transforms this value into device units and matches its absolute value against the character height of the available fonts.

For all height comparisons, the font mapper looks for the largest font that does not exceed the requested size.
This mapping occurs when the font is used for the first time.
(Read and Write property)

15.1.16   Italic as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether font is italic.
Notes: (Read and Write property)

15.1.17   LastError as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

15.1.18   LimitSize as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to limit font sizes.
Notes: ChooseFont should select only font sizes within the range specified by the MinSize and MaxSize members.
(Read and Write property)

15.1.19   MaxSize as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The maximum allowed font size.
Notes: Setting this also sets LimitSize to true.
(Read and Write property)

15.1.20   MinSize as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The minimum allowed font size.
Notes: Setting this also sets LimitSize to true.
(Read and Write property)

15.1.21   NoFontSimulations as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether ChooseFont should not display or allow selection of font simulations.
Notes: (Read and Write property)

15.1.22 NoInitialFaceSelection as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to show initial font selection.
Notes: When initializing the dialog box controls, use this flag to prevent the dialog box from displaying an initial selection for the font name combo box. This is useful when there is no single font name that applies to the text selection.
(Read and Write property)

15.1.23 NoInitialSizeSelection as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to show initialize size.
Notes: When initializing the dialog box controls, use this flag to prevent the dialog box from displaying an initial selection for the Font Size combo box. This is useful when there is no single font size that applies to the text selection.
(Read and Write property)

15.1.24 NoInitialStyleSelection as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to show initial style options.
Notes: When initializing the dialog box controls, use this flag to prevent the dialog box from displaying an initial selection for the Font Style combo box. This is useful when there is no single font style that applies to the text selection.
(Read and Write property)

15.1.25 NoVectorFonts as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether ChooseFont should not allow vector font selections.
Notes: (Read and Write property)

15.1.26 NoVerticalFonts as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether the Font dialog box lists only horizontally oriented fonts.
Notes: (Read and Write property)

15.1.27   OnlyFixedPitchFonts as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether ChooseFont should enumerate and allow selection of only fixed-pitch fonts.
Notes: (Read and Write property)

15.1.28   OnlyTrueTypeFonts as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether ChooseFont should only enumerate and allow the selection of TrueType fonts.
Notes: (Read and Write property)

15.1.29   Parent as Variant

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The parent window.
Notes: The window that owns the dialog box. This member can be any valid window, or it can be nil if the dialog box has no owner.
Can reference a Window or DesktopWindow object.
(Read and Write property)

15.1.30   ScalableFontsOnly as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to only allow scalable fonts.
Notes: Specifies that ChooseFont should allow only the selection of scalable fonts. Scalable fonts include vector fonts, scalable printer fonts, TrueType fonts, and fonts scaled by other technologies.
(Read and Write property)

15.1.31   ShowApply as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to show Apply button.
Notes: Causes the dialog box to display the Apply button.
Clicking Apply button does trigger Apply event.
(Read and Write property)

15.1.32 ShowInactiveFonts as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether to display hidden fonts.
Notes: ChooseFont can additionally display fonts that are set to Hide in Fonts Control Panel.
(Read and Write property)

15.1.33 Size as Double

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The size of the font.
Notes: (Read and Write property)

15.1.34 Strikethrough as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether font is stroke through.
Notes: (Read and Write property)

15.1.35 TextColor as Color

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The text color for the font.
Notes: Only used if Effects is true.
(Read and Write property)

15.1.36 Underline as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: Whether font is underlined.
Notes: (Read and Write property)

15.1.37 Weight as Integer

Plugin Version: 17.3, Platform: Windows, Targets: Desktop only.

Function: The weight of the font in the range 0 through 1000.
Notes: For example, 400 is normal and 700 is bold. If this value is zero, a default weight is used.

The following values are defined for convenience.

Value	Weight
FW_DONTCARE	0
FW_THIN	100
FW_EXTRALIGHT	200
FW_ULTRALIGHT	200
FW_LIGHT	300
FW_NORMAL	400
FW_REGULAR	400
FW_MEDIUM	500
FW_SEMIBOLD	600
FW_DEMIBOLD	600
FW_BOLD	700
FW_EXTRABOLD	800
FW_ULTRABOLD	800
FW_HEAVY	900
FW_BLACK	900

(Read and Write property)

15.1.38 Events

15.1.39 Apply

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The user pressed apply.
Notes: Our plugin automatically calls query for you to get current values from dialog into properties.

15.1.40   BoundsChanged

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The bounds changed.

15.1.41   BoundsChanging

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The bounds are changing.

15.1.42   GotFocus

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The window got focus.

15.1.43   Hide

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The window is hidden.

15.1.44   Init

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The dialog initialized.
Notes: You can work with dialog via DialogHandle property.

15.1.45   LostFocus

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The window lost focus.

15.1.46 Show

Plugin Version: 17.3, Platform: Windows, Targets: .

Function: The window is shown.

15.1.47 Constants

Font Types

Constant	Value	Description
FontTypeBold	&h100	The font weight is bold.
FontTypeItalic	&h200	The italic font attribute is set.
FontTypePrinter	&h4000	The font is a printer font.
FontTypeRegular	&h400	The font weight is normal.
FontTypeScreen	&h2000	The font is a screen font.
FontTypeSimulator	&h8000	The font is simulated by the graphics device interface (GDI).

15.2   class WindowsFontFamilyMBS

15.2.1   class WindowsFontFamilyMBS

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for details on Windows fonts.
Example:

dim lines(-1) as string

for each f as WindowsFontFamilyMBS in WindowsFontFamilyMBS.AllFonts
lines.append f.LogFontFullName
next

MsgBox Join(lines,EndOfLine)

Notes: The font list is the same as font() function in Xojo as far as we see.
Blog Entries

- MBS Xojo Plugins, version 21.4pr3

- News from the MBS Xojo Plugins Version 20.5

- MBS Xojo Plugins, version 20.5pr4

- MBS REALbasic Plugins 11.0 released

- MBS REALbasic Plugins, version 10.6pr12

15.2.2   Methods

15.2.3   AllFonts as WindowsFontFamilyMBS()

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all fonts on this windows machine.
Example:

dim lines(-1) as string

for each f as WindowsFontFamilyMBS in WindowsFontFamilyMBS.AllFonts
lines.append f.LogFontFullName
next

MsgBox Join(lines,EndOfLine)

Notes: Font list is queried from OS and not cached, so this should update if new fonts are activated.

Uses EnumFontFamiliesW function, see
https://docs.microsoft.com/en-us/windows/win32/api/wingdi/nf-wingdi-enumfontfamiliesw
See also:

15.2.4   AllFonts(fonts() as WindowsFontFamilyMBS) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all fonts on this windows machine.
Notes: Returns the number of fonts found and sets the font objects in the array.
if the array is too short, you get the first ubound(fonts)+1 fonts set there.

Font list is queried from OS and not cached, so this should update if new fonts are activated.
See also:

15.2.5   AllFontsEx as WindowsFontFamilyMBS()

Plugin Version: 20.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all fonts on this windows machine.
Example:

dim lines(-1) as string

for each f as WindowsFontFamilyMBS in WindowsFontFamilyMBS.AllFontsEx
lines.append f.LogFontFullName
next

MsgBox Join(lines,EndOfLine)

Notes: Font list is queried from OS and not cached, so this should update if new fonts are activated.

Uses EnumFontFamiliesExW function, see
https://docs.microsoft.com/en-us/windows/win32/api/wingdi/nf-wingdi-enumfontfamiliesexw

15.2.6 AxisMaxValue(index as Integer) as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum value for this axis.
Notes: Index is from zero to NumberOfAxes-1.
Only available for truetype fonts.

15.2.7 AxisMinValue(index as Integer) as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The minimum value for this axis.
Notes: Index is from zero to NumberOfAxes-1.
Only available for truetype fonts.

15.2.8 AxisName(index as Integer) as string

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the axis.
Notes: Index is from zero to NumberOfAxes-1.
Only available for truetype fonts.

15.2.9 DesignVectorValues(index as Integer) as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An array specifying the values of the axes of a multiple master OpenType font.
Notes: This array corresponds to the axes* arrays.

15.2.10 FontsOfFamily(family as string) as WindowsFontFamilyMBS()

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all fonts of the given font family on this windows machine.
See also:

15.2.11 FontsOfFamily(family as string, fonts() as WindowsFontFamilyMBS) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all fonts of the given font family on this windows machine.
Notes: Returns the number of fonts found and sets the font objects in the array.
if the array is too short, you get the first ubound(fonts)+1 fonts set there.
See also:

- 15.2.10 FontsOfFamily(family as string) as WindowsFontFamilyMBS() 456

15.2.12 Properties

15.2.13 CodepageBitfield as MemoryBlock

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: A 64-bit, code-page bitfield (CPB) that identifies a specific character set or code page. #
Notes: Code pages are in the lower 32 bits of this bitfield. The high 32 are used for non-Windows code pages. For more information, see Code Page Bitfields.
http://msdn.microsoft.com/en-us/library/dd317754(v=vs.85).aspx

Only available for truetype fonts.
(Read only property)

15.2.14 FontType as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The type of the font.
Notes: This parameter can be a combination of these values:

DEVICE_FONTTYPE
RASTER_FONTTYPE
TRUETYPE_FONTTYPE
(Read only property)

15.2.15 LogFontBold as Boolean

Plugin Version: 17.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether font is bold.
Notes: Setting to false, will set Weight to 400, setting to true will set Weight to 700.
Returns true if Weight is equal or more than 700.
(Read only property)

15.2.16 LogFontCharSet as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The character set. The following values are predefined.
Notes: The following values are predefined.

ANSI_CHARSET
BALTIC_CHARSET
CHINESEBIG5_CHARSET
DEFAULT_CHARSET
EASTEUROPE_CHARSET
GB2312_CHARSET
GREEK_CHARSET
HANGUL_CHARSET
MAC_CHARSET
OEM_CHARSET
RUSSIAN_CHARSET
SHIFTJIS_CHARSET
SYMBOL_CHARSET
TURKISH_CHARSET
VIETNAMESE_CHARSET

Korean language edition of Windows:

JOHAB_CHARSET

Middle East language edition of Windows:

ARABIC_CHARSET
HEBREW_CHARSET

Thai language edition of Windows:

THAI_CHARSET

The OEM_CHARSET value specifies a character set that is operating-system dependent.

DEFAULT_CHARSET is set to a value based on the current system locale. For example, when the system locale is English (United States), it is set as ANSI_CHARSET.

Fonts with other character sets may exist in the operating system. If an application uses a font with an unknown character set, it should not attempt to translate or interpret strings that are rendered with that font.

This parameter is important in the font mapping process. To ensure consistent results, specify a specific character set. If you specify a typeface name in the lfFaceName member, make sure that the lfCharSet value matches the character set of the typeface specified in lfFaceName.
(Read only property)

15.2.17   LogFontClipPrecision as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The clipping precision.
Notes: The clipping precision defines how to clip characters that are partially outside the clipping region. It can be one or more of the following values.

For more information about the orientation of coordinate systems, see the description of the nOrientation parameter.

Value	Meaning
CLIP_CHARACTER_PRECIS	Not used.
CLIP_DEFAULT_PRECIS	Specifies default clipping behavior.
CLIP_DFA_DISABLE	Windows XP SP1: Turns off font association for the font. Note that this flag is not guaranteed to have any effect on any platform after Windows Server 2003.
CLIP_EMBEDDED	You must specify this flag to use an embedded read-only font.
CLIP_LH_ANGLES	When this value is used, the rotation for all fonts depends on whether the orientation of the coordinate system is left-handed or right-handed. If not used, device fonts always rotate counterclockwise, but the rotation of other fonts is dependent on the orientation of the coordinate system.
CLIP_MASK	Not used.
CLIP_DFA_OVERRIDE	Turns off font association for the font. This is identical to CLIP_DFA_DISABLE, but it can have problems in some situations; the recommended flag to use is CLIP_DFA_DISABLE.
CLIP_STROKE_PRECIS	Not used by the font mapper, but is returned when raster, vector, or TrueType fonts are enumerated. For compatibility, this value is always returned when enumerating fonts.
CLIP_TT_ALWAYS	Not used.

(Read only property)

15.2.18   LogFontEscapement as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The angle, in tenths of degrees, between the escapement vector and the x-axis of the device.
Notes: The escapement vector is parallel to the base line of a row of text.

When the graphics mode is set to GM_ADVANCED, you can specify the escapement angle of the string independently of the orientation angle of the string's characters.

When the graphics mode is set to GM_COMPATIBLE, LogFontEscapement specifies both the escapement and orientation. You should set LogFontEscapement and LogFontOrientation to the same value.
(Read only property)

15.2.19   LogFontFaceName as String

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the typeface name of the font.
Notes: (Read only property)

15.2.20   LogFontFullName as String

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The unique name of the font.
Notes: For example, ABC Font Company TrueType Bold Italic Sans Serif.
(Read only property)

15.2.21   LogFontHeight as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height, in logical units, of the font's character cell or character.
Notes: The character height value (also known as the em height) is the character cell height value minus the internal-leading value. The font mapper interprets the value specified in lfHeight in the following manner.

For all height comparisons, the font mapper looks for the largest font that does not exceed the requested size.

Value	Meaning
>0	The font mapper transforms this value into device units and matches it against the cell height of the available fonts.
0	The font mapper uses a default height value when it searches for a match.
<0	The font mapper transforms this value into device units and matches its absolute value against the character height of the available fonts.

This mapping occurs when the font is used for the first time.
(Read only property)

15.2.22   LogFontItalic as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An italic font if set to true.
Notes: (Read only property)

15.2.23   LogFontOrientation as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The angle, in tenths of degrees, between each character's base line and the x-axis of the device.
Notes: (Read only property)

15.2.24   LogFontOutPrecision as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The output precision.
Notes: The output precision defines how closely the output must match the requested font's height, width, character orientation, escapement, pitch, and font type. It can be one of the following values.

Applications can use the OUT_DEVICE_PRECIS, OUT_RASTER_PRECIS, OUT_TT_PRECIS, and OUT_PS_ONLY_PRECIS values to control how the font mapper chooses a font when the operating system contains more than one font with a specified name. For example, if an operating system contains a font named Symbol in raster and TrueType form, specifying OUT_TT_PRECIS forces the font mapper to choose the TrueType version. Specifying OUT_TT_ONLY_PRECIS forces the font mapper to choose a TrueType font, even if it must substitute a TrueType font of another name.
(Read only property)

Value	Meaning
OUT_CHARACTER_PRECIS	Not used.
OUT_DEFAULT_PRECIS	Specifies the default font mapper behavior.
OUT_DEVICE_PRECIS	Instructs the font mapper to choose a Device font when the system contains multiple fonts with the same name.
OUT_OUTLINE_PRECIS	This value instructs the font mapper to choose from TrueType and other outline-based fonts.
OUT_PS_ONLY_PRECIS	Instructs the font mapper to choose from only PostScript fonts. If there are no PostScript fonts installed in the system, the font mapper returns to default behavior.
OUT_RASTER_PRECIS	Instructs the font mapper to choose a raster font when the system contains multiple fonts with the same name.
OUT_STRING_PRECIS	This value is not used by the font mapper, but it is returned when raster fonts are enumerated.
OUT_STROKE_PRECIS	This value is not used by the font mapper, but it is returned when TrueType, other outline-based fonts, and vector fonts are enumerated.
OUT_TT_ONLY_PRECIS	Instructs the font mapper to choose from only TrueType fonts. If there are no TrueType fonts installed in the system, the font mapper returns to default behavior.
OUT_TT_PRECIS	Instructs the font mapper to choose a TrueType font when the system contains multiple fonts with the same name.

15.2.25   LogFontPitchAndFamily as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The pitch and family of the font.
Notes: The two low-order bits specify the pitch of the font and can be one of the following values.

DEFAULT_PITCH
FIXED_PITCH
VARIABLE_PITCH

Bits 4 through 7 of the member specify the font family and can be one of the following values.

FF_DECORATIVE
FF_DONTCARE
FF_MODERN
FF_ROMAN
FF_SCRIPT
FF_SWISS

The proper value can be obtained by using the Boolean OR operator to join one pitch constant with one

family constant.

Font families describe the look of a font in a general way. They are intended for specifying fonts when the exact typeface desired is not available. The values for font families are as follows.

Value	Meaning
FF_DECORATIVE	Novelty fonts. Old English is an example.
FF_DONTCARE	Use default font.
FF_MODERN	Fonts with constant stroke width (monospace), with or without serifs. Monospace fonts are usually modern. Pica, Elite, and CourierNew are examples.
FF_ROMAN	Fonts with variable stroke width (proportional) and with serifs. MS Serif is an example.
FF_SCRIPT	Fonts designed to look like handwriting. Script and Cursive are examples.
FF_SWISS	Fonts with variable stroke width (proportional) and without serifs. MS Sans Serif is an example.

(Read only property)

15.2.26 LogFontQuality as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The output quality.
Notes: The output quality defines how carefully the graphics device interface (GDI) must attempt to match the logical-font attributes to those of an actual physical font. It can be one of the following values.

(Read only property)

15.2.27 LogFontScript as String

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The script, that is, the character set, of the font. For example, Cyrillic.
Notes: (Read only property)

15.2.28 LogFontStrikeOut as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Value	Meaning
ANTIALIASED_QUALITY	Font is always antialiased if the font supports it and the size of the font is not too small or too large.
CLEARTYPE_QUALITY	If set, text is rendered (when possible) using ClearType antialiasing method. See Remarks for more information.
DEFAULT_QUALITY	Appearance of the font does not matter.
DRAFT_QUALITY	Appearance of the font is less important than when PROOF_QUALITY is used. For GDI raster fonts, scaling is enabled, which means that more font sizes are available, but the quality may be lower. Bold, italic, underline, and strikeout fonts are synthesized if necessary.
NONANTIALIASED_QUALITY	Font is never antialiased.
PROOF_QUALITY	Character quality of the font is more important than exact matching of the logical-font attributes. For GDI raster fonts, scaling is disabled and the font closest in size is chosen. Although the chosen font size may not be mapped exactly when PROOF_QUALITY is used, the quality of the font is high and there is no distortion of appearance. Bold, italic, underline, and strikeout fonts are synthesized if necessary.

Function: A strikeout font if set to true.
Notes: (Read only property)

15.2.29   LogFontStyle as String

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The style of the font. For example, Bold Italic.
Notes: (Read only property)

15.2.30   LogFontUnderline as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An underlined font if set to true.
Notes: (Read only property)

15.2.31   LogFontWeight as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The weight of the font in the range 0 through 1000.
Notes: For example, 400 is normal and 700 is bold. If this value is zero, a default weight is used.

The following values are defined for convenience.

Value	Weight
FW_DONTCARE	0
FW_THIN	100
FW_EXTRALIGHT	200
FW_ULTRALIGHT	200
FW_LIGHT	300
FW_NORMAL	400
FW_REGULAR	400
FW_MEDIUM	500
FW_SEMIBOLD	600
FW_DEMIBOLD	600
FW_BOLD	700
FW_EXTRABOLD	800
FW_ULTRABOLD	800
FW_HEAVY	900
FW_BLACK	900

(Read only property)

15.2.32   LogFontWidth as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The average width, in logical units, of characters in the font.
Notes: If LogFontWidth is zero, the aspect ratio of the device is matched against the digitization aspect ratio of the available fonts to find the closest match, determined by the absolute value of the difference.
(Read only property)

15.2.33   NumberOfAxes as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of axes for a specified multiple master font.
Notes: The axes arrays contains information on all the axes of a multiple master font.
Only available for truetype fonts.
(Read only property)

15.2.34   NumberOfDesignVectors as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The design vectors are used by an application to specify values for the axes of a multiple master font.
Notes: (Read only property)

15.2.35   TextMetricAscent as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ascent (units above the base line) of characters.
Notes: (Read only property)

15.2.36   TextMetricAverageCharWidth as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The average width of characters in the font (generally defined as the width of the letter x).
Notes: This value does not include overhang required for bold or italic characters.
(Read only property)

15.2.37   TextMetricAverageWidth as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The average width of characters in the font, in notional units.
Notes: This value should be compared with the value of the TextMetricSizeEM member.
Only available for truetype fonts.
(Read only property)

15.2.38   TextMetricBreakChar as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The value of the character to be used to define word breaks for text justification.
Notes: (Read only property)

15.2.39   TextMetricCellHeight as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height, in notional units, of the font.
Notes: This value should be compared with the value of the TextMetricSizeEM member.
Only available for truetype fonts.
(Read only property)

15.2.40   TextMetricCharSet as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The character set of the font.
Notes: (Read only property)

15.2.41   TextMetricDefaultChar as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The value of the character to be substituted for characters that are not in the font.
Notes: (Read only property)

15.2.42   TextMetricDescent as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The descent (units below the base line) of characters.
Notes: (Read only property)

15.2.43   TextMetricDigitizedAspectX as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The horizontal aspect of the device for which the font was designed.
Notes: (Read only property)

15.2.44   TextMetricDigitizedAspectY as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The vertical aspect of the device for which the font was designed.
Notes: The ratio of the tmDigitizedAspectX and tmDigitizedAspectY members is the aspect ratio of the device for which the font was designed.
(Read only property)

15.2.45   TextMetricExternalLeading as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The amount of extra leading (space) that the application adds between rows.
Notes: Since this area is outside the font, it contains no marks and is not altered by text output calls in either OPAQUE or TRANSPARENT mode. The designer may set this member to zero.
(Read only property)

15.2.46   TextMetricFirstChar as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The value of the first character defined in the font.
Notes: (Read only property)

15.2.47   TextMetricFlags as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies whether the font is italic, underscored, outlined, bold, and so forth.
Notes: May be any reasonable combination of the following values.

Bit	Name	Meaning
0	NTM_ITALIC	italic
5	NTM_BOLD	bold
8	NTM_REGULAR	regular
16	NTM_NONNEGATIVE_AC	no glyph in a font at any size has a negative A or C space.
17	NTM_PS_OPENTYPE	PostScript OpenType font
18	NTM_TT_OPENTYPE	TrueType OpenType font
19	NTM_MULTIPLEMASTER	multiple master font
20	NTM_TYPE1	Type 1 font
21	NTM_DSIG	font with a digital signature. This allows traceability and ensures that the font has been tested and is not corrupted

Only available for truetype fonts.
(Read only property)

15.2.48 TextMetricHeight as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height (ascent + descent) of characters.
Notes: (Read only property)

15.2.49 TextMetricInternalLeading as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The amount of leading (space) inside the bounds set by the tmHeight member.
Notes: Accent marks and other diacritical characters may occur in this area. The designer may set this member to zero.
(Read only property)

15.2.50 TextMetricItalic as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An italic font if set to true.
Notes: (Read only property)

15.2.51 TextMetricLastChar as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The value of the last character defined in the font.
Notes: (Read only property)

15.2.52 TextMetricMaxCharWidth as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the widest character in the font.
Notes: (Read only property)

15.2.53 TextMetricOverhang as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The extra width per string that may be added to some synthesized fonts.
Notes: When synthesizing some attributes, such as bold or italic, graphics device interface (GDI) or a device may have to add width to a string on both a per-character and per-string basis. For example, GDI makes a string bold by expanding the spacing of each character and overstriking by an offset value; it italicizes a font by shearing the string. In either case, there is an overhang past the basic string. For bold strings, the overhang is the distance by which the overstrike is offset. For italic strings, the overhang is the amount the top of the font is sheared past the bottom of the font.

The TextMetricOverhang member enables the application to determine how much of the character width returned by a GetTextExtentPoint32 function call on a single character is the actual character width and how much is the per-string extra width. The actual width is the extent minus the overhang.
(Read only property)

15.2.54   TextMetricPitchAndFamily as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The pitch and family of the selected font.
Notes: The low-order bit (bit 0) specifies the pitch of the font. If it is 1, the font is variable pitch (or proportional). If it is 0, the font is fixed pitch (or monospace). Bits 1 and 2 specify the font type. If both bits are 0, the font is a raster font; if bit 1 is 1 and bit 2 is 0, the font is a vector font; if bit 1 is 0 and bit 2 is set, or if both bits are 1, the font is some other type. Bit 3 is 1 if the font is a device font; otherwise, it is 0.

The four high-order bits designate the font family. The TextMetricPitchAndFamily member can be combined with the hexadecimal value 0xF0 by using the bitwise AND operator and can then be compared with the font family names for an identical match. For more information about the font families, see LogFont-PitchAndFamily.
(Read only property)

15.2.55   TextMetricSizeEM as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the em square for the font.
Notes: This value is in notional units (that is, the units for which the font was designed).
Only available for truetype fonts.
(Read only property)

15.2.56   TextMetricStruckOut as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An strikeout font if set to true.
Notes: (Read only property)

15.2.57   TextMetricUnderlined as Boolean

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: An underline font if set to true.
Notes: (Read only property)

15.2.58   TextMetricWeight as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The weight of the font.
Notes: (Read only property)

15.2.59   UnicodeSubsetBitfield as MemoryBlock

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: A 128-bit Unicode subset bitfield (USB) identifying up to 126 Unicode subranges.
Notes: Each bit, except the two most significant bits, represents a single subrange. The most significant bit is always 1 and identifies the bitfield as a font signature; the second most significant bit is reserved and must be 0. Unicode subranges are numbered in accordance with the ISO 10646 standard. For more information, see Unicode Subset Bitfields.
http://msdn.microsoft.com/en-us/library/dd374090(v=vs.85).aspx

Only available for truetype fonts.
(Read only property)

15.2.60   Constants

Constants

Constant	Value	Description
ANSI_CHARSET	0	One of the constants for the LogFontCharSet property.
ANTIALIASED_QUALITY	4	One of the constants for the LogFontQuality property. Enables antialiasing for the font. The display driver must suppo text for this setting to work.
ARABIC_CHARSET	178	One of the constants for the LogFontCharSet property.
BALTIC_CHARSET	186	One of the constants for the LogFontCharSet property.
CHINESEBIG5_CHARSET	136	One of the constants for the LogFontCharSet property.
CLIP_CHARACTER_PRECIS	1	One of the constants for the LogFontClipPrecision property. Not used.
CLIP_DEFAULT_PRECIS	0	One of the constants for the LogFontClipPrecision property. Specifies default clipping behavior.
CLIP_EMBEDDED	128	One of the clip constants.
CLIP_LH_ANGLES	16	One of the clip constants.
CLIP_MASK	&h15	One of the clip constants.
CLIP_STROKE_PRECIS	2	One of the constants for the LogFontClipPrecision property. Not used by the font mapper, but is returned when raster, vector, fonts are enumerated.
CLIP_TT_ALWAYS	32	One of the clip constants.
DEFAULT_CHARSET	1	One of the constants for the LogFontCharSet property.
DEFAULT_PITCH	0	One of the pitch constants.
DEFAULT_QUALITY	0	One of the constants for the LogFontQuality property. Appearance of the font does not matter.
DEVICE_FONTTYPE	&h002	One of the constants for the FontType property.
DRAFT_QUALITY	1	One of the constants for the LogFontQuality property. For GDI raster fonts, scaling is enabled, which means that more available, but the quality may be lower. Bold, italic, underline, fonts are synthesized if necessary.
EASTEUROPE_CHARSET	238	One of the constants for the LogFontCharSet property.
FF_DECORATIVE	80	One of the constants for the LogFontPitchAndFamily property. Novelty fonts, for example, Old English.
FF_DONTCARE	0	One of the constants for the LogFontPitchAndFamily property. Do not care or do not know.
FF_MODERN	48	One of the constants for the LogFontPitchAndFamily property. Fonts with constant stroke width (monospace), with or wi Monospace fonts are usually modern, for example, Pica, Elite, New.
FF_ROMAN	16	One of the constants for the LogFontPitchAndFamily property. Fonts with variable stroke width (proportional) and with serifs, Serif.
FF_SCRIPT	64	One of the constants for the LogFontPitchAndFamily property. Fonts designed to look like handwriting, for example, Script and
FF_SWISS	32	One of the constants for the LogFontPitchAndFamily property. Fonts with variable stroke width (proportional) and without serifs, Sans Serif.
FIXED_PITCH	1	One of the pitch constants.
FW_BLACK	900	One of the constants for the LogFontWeight property.
FW_BOLD	700	One of the constants for the LogFontWeight property.
FW_DEMIBOLD	600	One of the constants for the LogFontWeight property.
FW_DONTCARE	0	One of the constants for the LogFontWeight property.
FW_EXTRABOLD	800	One of the constants for the LogFontWeight property.
FW_EXTRALIGHT	200	One of the constants for the LogFontWeight property.
FW_HEAVY	900	One of the constants for the LogFontWeight property.
FW_LIGHT	300	One of the constants for the LogFontWeight property.
FW_MEDIUM	500	One of the constants for the LogFontWeight property.
FW_NORMAL	400	One of the constants for the LogFontWeight property.
FW_REGULAR	400	One of the constants for the LogFontWeight property.
FW_SEMIBOLD	600	One of the constants for the LogFontWeight property.
FW_THIN	100	One of the constants for the LogFontWeight property.

Chapter 16

Graphics & Pictures

16.1 class Graphics

16.1.1 class Graphics

Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends Xojo's Graphics Class.

16.1.2 Methods

16.1.3 DrawWindowsIconMBS(file as folderitem, IconID as Integer, x as Integer, y as Integer, w as Integer, h as Integer) as boolean

Plugin Version: 10.1, Platform: Windows, Targets: Desktop only.

Function: Draws a windows icon on the given rectangle.
Notes: The folderitem should point to a file with icons (exe, dll, ico or something else). If IconID is zero, the first icon is used. Else IconID is the id of the icon.
If w or h is 0, the default size is used.
Returns true on success and false on failure.

16.1.4 StretchBltMBS(nXOriginDest as Integer, nYOriginDest as Integer, nWidthDest as Integer, nHeightDest as Integer, source as graphics, nXOriginSrc as Integer, nYOriginSrc as Integer, nWidthSrc as Integer, nHeightSrc as Integer, dwRop as Integer) as boolean

Plugin Version: 6.1, Platform: Windows, Targets: Desktop only.

Function: The StretchBlt function copies a bitmap from a source rectangle into a destination rectangle, stretching or compressing the bitmap to fit the dimensions of the destination rectangle, if necessary.
Notes: This is just a wrapper to the StretchBlt function from the Windows API.

The system stretches or compresses the bitmap according to the stretching mode currently set in the destination device context.

Parameters:

nXOriginDest	Specifies the x-coordinate, in logical units, of the upper-left corner of the destination rectangle.
nYOriginDest	Specifies the y-coordinate, in logical units, of the upper-left corner of the destination rectangle.
nWidthDest	Specifies the width, in logical units, of the destination rectangle.
nHeightDest	Specifies the height, in logical units, of the destination rectangle.
hdcSrc	The source device context.
nXOriginSrc	Specifies the x-coordinate, in logical units, of the upper-left corner of the source rectangle.
nYOriginSrc	Specifies the y-coordinate, in logical units, of the upper-left corner of the source rectangle.
nWidthSrc	Specifies the width, in logical units, of the source rectangle.
nHeightSrc	Specifies the height, in logical units, of the source rectangle.
dwRop	Specifies the raster operation to be performed. Raster operation codes define how the system combines colors in output operations that involve a brush, a source bitmap, and a destination bitmap. See BitBlt for a list of common raster operation codes (ROPs). Note that the CAPTUREBLT ROP generally cannot be used for printing device contexts.

Return Values

If the function succeeds, the return value is true.
If the function fails, the return value is false.

Possible operation modes:

SRCCOPY	&h00CC0020	dest = source	Copies the source rectangle directly to the destination rectangle.
SRCPAINT	&h00EE0086	dest = source OR dest	Combines the colors of the source and destination rectangles by using the Boolean OR operator.
SRCAND	&h008800C6	dest = source AND dest	Combines the colors of the source and destination rectangles by using the Boolean AND operator.
SRCINVERT	&h00660046	dest = source XOR dest	Combines the colors of the source and destination rectangles by using the Boolean XOR operator.
SRCERASE	&h00440328	dest = source AND (NOT dest)	Combines the inverted colors of the destination rectangle with the colors of the source rectangle by using the Boolean AND operator.
NOTSRCCOPY	&h00330008	dest = (NOT source)	Copies the inverted source rectangle to the destination.
NOTSRCERASE	&h001100A6	dest = (NOT src) AND (NOT dest)	Combines the colors of the source and destination rectangles by using the Boolean OR operator and then inverts the resultant color.
MERGECOPY	&h00C000CA	dest = (source AND pattern)	Merges the colors of the source rectangle with the brush currently selected in hdcDest, by using the Boolean AND operator.
MERGEPAINT	&h00BB0226	dest = (NOT source) OR dest	Merges the colors of the inverted source rectangle with the colors of the destination rectangle by using the Boolean OR operator.
PATCOPY	&h00F00021	dest = pattern	Copies the brush currently selected in hdcDest, into the destination bitmap.
PATPAINT	&h00FB0A09	dest = DPSnoo	Combines the colors of the brush currently selected in hdcDest, with the colors of the inverted source rectangle by using the Boolean OR operator. The result of this operation is combined with the colors of the destination rectangle by using the Boolean OR operator.
PATINVERT	&h005A0049	dest = pattern XOR dest	Combines the colors of the brush currently selected in hdcDest, with the colors of the destination rectangle by using the Boolean XOR operator.
DSTINVERT	&h00550009	dest = (NOT dest)	Inverts the destination rectangle.
BLACKNESS	&h00000042	dest = BLACK	Fills the destination rectangle using the color associated with index 0 in the physical palette. (This color is black for the default physical palette.)
WHITENESS	&h00FF0062	dest = WHITE	Fills the destination rectangle using the color associated with index 1 in the physical palette. (This color is white for the default physical palette.)
NOMIRRORBITMAP	&h80000000	Do not Mirror the bitmap in this call.	Prevents the bitmap from being mirrored.
CAPTUREBLT	&h40000000	Include layered windows.	Includes any windows that are layered on top of your window in the resulting image. By default, the image only contains your window. Note that this generally cannot be used for printing device contexts.

16.1.5 WinApplyDevModeMBS(devmode as WindowsDeviceModeMBS) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Updates the specified printer or plotter device context (graphics) using the specified information.
Notes: Returns true on success.

This function cannot be used to change the driver name, device name, or the output port. When the user changes the port connection or device name, the application must delete the original graphics and create a new graphics object with the new information.

Windows seems not to allow changing graphics object for printer while a page is open, so use WinEnd-PageMBS to close page before changing settings.

Does not work for Xojo 2016r4 and newer due to switch to DirectDraw for printing.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr6

- More on printing on Windows

16.1.6　WindowsGraphicsInfoMBS as WindowsGraphicsInfoMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Query information about graphics object on Windows.
Notes: Returns nil on any error.

16.1.7　WinEndPageMBS as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The EndPage function notifies the device that the application has finished writing to a page.
Notes: This function is typically used to direct the device driver to advance to a new page.

Returns true on success.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation-factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

Use the WinApplyDevModeMBS function to change the device mode, if necessary, after calling the EndPage function. Note that a call to WinApplyDevModeMBS resets all device context attributes back to default values. Neither EndPage nor StartPage resets the device context attributes. Device context attributes remain constant across subsequent pages. You do not need to re-select objects and set up the mapping mode again before printing the next page; however, doing so will produce the same results and reduce code differences between versions of Windows.

When a page in a spooled file exceeds approximately 350 MB, it may fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.

Does not work for Xojo 2016r4 and newer due to switch to DirectDraw for printing.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr6

- More on printing on Windows

16.1.8　WinStartPageMBS as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The StartPage function prepares the printer driver to accept data.
Notes: Returns true on success.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation-factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

The system disables the WinApplyDevModeMBS function between calls to the StartPage and EndPage functions. This means that you cannot change the device mode except at page boundaries. After calling EndPage, you can call WinApplyDevModeMBS to change the device mode, if necessary. Note that a call to WinApplyDevModeMBS resets all device context attributes back to default values.

Neither EndPage nor StartPage resets the device context attributes. Device context attributes remain constant across subsequent pages. You do not need to re-select objects and set up the mapping mode again before printing the next page; however, doing so will produce the same results and reduce code differences between versions of Windows.

Does not work for Xojo 2016r4 and newer due to switch to DirectDraw for printing.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr6

- More on printing on Windows

16.2 Globals

16.2.1 WindowsDrawPictureIntoDeviceContextMBS(pic as picture, HDC as Integer, x as Integer, y as Integer, w as Integer, h as Integer, Transparent as boolean)

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Draws a Xojo Picture into a HDC on Windows.
Example:

```
// get a picture
dim logo as Picture = LogoMBS(500)

// get HDC
dim h as Integer = g.Handle(g.HandleTypeHDC)

// and draw into it
WindowsDrawPictureIntoDeviceContextMBS(logo, h, 0, 0, 500, 500, false)
```

Notes: Some SDKs from other companies give you sometimes HDC value to draw your stuff inside. You can prepare a Xojo picture and copy it into the HDC.
Specify rectangle and whether to use transparency.

Chapter 17

HTMLViewer

17.1 class DesktopHTMLViewer

17.1.1 class DesktopHTMLViewer

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Class in Xojo for HTML rendering.
Notes: We have multiple methods for macOS, Windows and Linux, so please check each method.

17.1.2 Methods

17.1.3 ChromiumBrowserMBS as ChromiumBrowserMBS

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Queries browser object for htmlviewer.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox b.MainFrame.URL

Notes: Works only on Windows if Webkit is chosen as renderer.
Returns nil on any error.

Please call all chromium functions only on main thread.
Do not call in open event of HTMLViewer as browser control is not yet initialized there.

Chapter 18

HTMLViewer Linux

18.1   class HTMLViewer

18.1.1   class HTMLViewer

Plugin Version: 7.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Class in Xojo 2005 for HTML rendering.
Notes: We have multiple methods for macOS, Windows and Linux, so please check each method.

18.1.2   Methods

18.1.3   ChromiumBrowserMBS as ChromiumBrowserMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Queries browser object for htmlviewer.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox b.MainFrame.URL

Notes: Works only on Windows if Webkit is chosen as renderer.
Returns nil on any error.

Please call all chromium functions only on main thread.
Do not call in open event of HTMLViewer as browser control is not yet initialized there.

May not work in Xojo 2021r3 due to changes inside the control.

18.1.4   IEContinueFindTextMBS(text as string, count as Integer, flags as Integer, selectText as boolean) as boolean

Plugin Version: 7.7, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Continues a search started with IEFindTextMBS.
Notes: Parameters are the same as for IEFindTextMBS.

18.1.5   IEDocumentMBS as IEDocumentMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries web document object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.

Not available for DesktopHTMLViewer control since there is no support for Internet Explorer for that control.

18.1.6   IEDrawToHDCMBS(HDC as Ptr, PrinterName as string = """) as boolean

Plugin Version: 12.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Draws the content of the html document into the graphics context.
Notes: As of Windows Internet Explorer 9, this method is deprecated and should not be used.
Returns true on success and false on failure.

With some printers, running DrawToDC may cause problems. You can ensure that DrawToDC works properly on all printers by running SetDocumentPrinter method first, and then passing the modified device context to DrawToDC. The plugin calls SetDocumentPrinter for you when you provide a printer name.

18.1.7   IEFileCreationDateMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the date the file was created.

Notes: Example value: "09/13/2007"

18.1.8 IEFileModifiedDateMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the date the file was last modified.
Notes: Example value: "12/03/2007"

18.1.9 IEFileSizeMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the file size.
Notes: Example value: "12475"

18.1.10 IEFileUpdatedDateMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the date the file was last updated.
Notes: Example value: "01/01/1601"

18.1.11 IEFindTextMBS(text as string, count as Integer, flags as Integer, selectText as boolean) as boolean

Plugin Version: 7.7, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Finds text on the current website.
Notes: text: the string that specifies the text to find.
count: long that specifies the number of characters to search from the starting point of the range. A positive integer indicates a forward search; a negative integer indicates a backward search.
Flags: integer that specifies one or more of the following flags to indicate the type of search:

Returns true: The search text was found.
Returns false: The search text was not found.

0	Default. Match partial words.
1	Match backwards.
2	Match whole words only.
4	Match case.
131072	Match bytes.
536870912	Match diacritical marks.
1073741824	Match Kashida character.
2147483648	Match AlefHamza character.

18.1.12   IEGetTextAreaMBS(FormName as String, FieldName as String) as String

Plugin Version: 16.3, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Queries text for a textarea.
Notes: FormName can be ”” to look for any field with given name.
Raises exception if field is not found.
Returns text from textarea.
Blog Entries

- HTMLViewer JavaScript communication for Xojo

- HTMLViewer JavaScript communication for Xojo

18.1.13   IEHandleMBS as Integer

Plugin Version: 8.0, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** The handle to the windows browser object.
Notes: If this value is zero, the plugin htmlviewer functions for Windows will not work.

18.1.14   IEHistoryBackMBS

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEHistoryMBS class instead. **Function:** Loads a previous URL from the History list.

18.1.15 IEHistoryForwardMBS

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEHistoryMBS class instead. **Function:** Loads the next URL from the History list.

18.1.16 IEHistoryLengthMBS as Integer

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEHistoryMBS class instead. **Function:** Retrieves the number of elements in the History list.
Notes: Example value: "0"

18.1.17 IEHTMLTextMBS as string

Plugin Version: 7.7, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Returns a copy of the html source code of the current webpage.
Example:

msgbox htmlviewer1.IEHTMLTextMBS

Notes: Improved in plugin version 12.2 to return better HTML text. This is the html generated from current web content and not the page we originally loaded. So this works with IEEditableMBS property. Returns "" on any error.
Blog Entries

- MBS Xojo Plugins, version 20.2pr3
- MBS Real Studio Plugins, version 12.2pr1

18.1.18 IEImageMBS as picture

Plugin Version: 9.5, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Copies the picture from the htmlviewer.
Example:

Sub Action()
ClearFocus

Dim picWeb As Picture

// Get image from IE
picWeb = HTMLViewer1.IEImageMBS

canvas1.backdrop = picWeb
End Sub

Notes: You may want to resize the htmlviewer to get a picture without scrollbars. (See example projects) You may need to call ClearFocus as it seems like if the focus is on the htmlviewer it does not draw itself in our picture.
Blog Entries

- Picture from HTMLViewer

- MBS REALbasic Plugins, version 10.6pr5

- Render Image from a HTMLViewer

- MBS REALbasic plug-ins version 9.5

18.1.19   IELastModifiedMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Gets the date that the page was last modified, if the page supplies one. **Notes:** Example value: "12/03/2007 20:08:17"

18.1.20   IELoadHTMLMBS(HTMLText as string) as boolean

Plugin Version: 15.1, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Loads the HTML text into the htmlviewer.
Example:

// in window open event, load blank page
HTMLViewer1.LoadURL "about:blank"

// later somewhere in app load HTML:
HTMLViewer1.IELoadHTMLMBS "<html><body>Hello World</body></html>"

Notes: Does not use a temp file like Xojo's built in method.

Returns true on success.

On Windows you may need to reset webviewer before or load "about:blank" to initialize the webviewer by Xojo (or Xojo).

Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr5

18.1.21 IEMimeTypeMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the MIME type for the file.
Notes: Example value: "HTM-Datei"
See also MimeTypeToFileExtensionMBS function.

18.1.22 IENamePropMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Gets the title of the document file.
Notes: Example value: "Apple"

18.1.23 IENavigatorAppMinorVersionMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves the application's minor version value.
Notes: Example value: ";SP2;"

18.1.24 IENavigatorAppNameMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves the name of the browser.
Notes: Example value: "Microsoft Internet Explorer"

18.1.25   IENavigatorAppVersionMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves the platform and version of the browser.
Notes: Example value: "4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 2.0.50727)"

18.1.26   IENavigatorBrowserLanguageMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves the current browser language.
Notes: Example value: "de"

18.1.27   IENavigatorCookieEnabledMBS as boolean

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves whether client-side persistent cookies are enabled in the browser.
Notes: Persistent cookies are those that are stored on the client-side computer.
Example value: "True"

18.1.28   IENavigatorJavaEnabledMBS as boolean

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Returns whether Java is enabled.
Notes: Example value: "True"

18.1.29   IENavigatorOnLineMBS as boolean

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves a value indicating whether the system is in global offline mode.
Notes: The user can modify the global offline state by choosing Work Offline from the File menu in Microsoft Internet Explorer version 4.0 or later. This property does not indicate whether the system is connected to

the network.

Example value: "True"

18.1.30 IENavigatorUserAgentMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves a string equivalent to the HTTP user-agent request header. **Notes:** The HTTP user-agent request header contains information about compatibility, the browser, and the platform name. For more information about the browser, see the IENavigatorappNameMBS property. For more information about the platform, see the IENavigatorappVersionMBS property.

The IENavigatoruserAgentMBS property dynamically returns a different value depending on the browser and platform versions. For example, Microsoft Internet Explorer 6 returns the following string for Microsoft Windows XP.

Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)

Example value: "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 2.0.50727)"

Like other plugin function, this can only work if a page has been loaded. Xojo won't create the internal htmlviewer object before you load a page.

18.1.31 IENavigatorUserLanguageMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IENavigatorMBS class instead. **Function:** Retrieves the operating system's natural language setting.
Notes: Example value: "de"

18.1.32 IEPrintMBS as boolean

Plugin Version: 9.5, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEWindowMBS class instead. **Function:** Opens the normal print dialog for the Internet Explorer.

Notes: Returns false on failure.
Blog Entries

- MBS REALbasic plug-ins version 9.5

18.1.33 IEPrintPreviewMBS as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Commands Internet Explorer to show the print preview dialog for this htmlviewer.
Example:

call htmlviewer1.IEPrintPreviewMBS

Notes: Returns true on success. Returns false if function is not supported.
The function returns directly while the preview dialog is still running.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr6

18.1.34 IEProtocolMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or retrieves the protocol portion of a URL.
Notes: Example value: "HTTP (HyperText Transfer-Protokoll)"

18.1.35 IEReadyStateMBS as string

Plugin Version: 12.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves a value that indicates the current state of the htmlviewer.
Notes: uninitialized: Object is not initialized with data.
loading: Object is loading its data.
loaded: Object has finished loading its data.
interactive: User can interact with the object even though it is not fully loaded.
complete: Object is completely initialized.

An object's state is initially set to uninitialized, and then to loading. When data loading is complete, the state of the link object passes through the loaded and interactive states to reach the complete state.

The states through which an object passes are determined by that object; an object can skip certain states (for example, interactive) if the state does not apply to that object.

Data source objects and databound elements are normally populated asynchronously, and certain programmatic operations can only be performed reliably on databound objects when they are ready for use. Therefore, the appropriate code should be written to confirm the readyState of objects prior to performing certain operations on them. For example, walking the rows of a table should not be attempted until after the table has reached the complete state.

The readyState property enables the status of an object to be tested. The correct place to test the readyState property is in the event handler for onreadystatechange. Similarly, a data source object (DSO) fires the ondatasetcomplete event to notify the document that the dataset is ready for programmatic operation.
Blog Entries

- MBS Real Studio Plugins, version 12.2pr1

18.1.36 IERefCountMBS as Integer

Plugin Version: 15.1, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Queries the reference count of the IE browser object.
Notes: Useful to check if references are leaked.

18.1.37 IEReferrerMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Gets the URL of the location that referred the user to the current page.
Notes: Example value: "http://www.apple.com/"

18.1.38 IEReloadMBS(Force as boolean = false) as boolean

Plugin Version: 12.0, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Reloads the current page.
Notes: Boolean that specifies one of the following possible values:

False: Default. Reloads the document from the cache.
True: Reloads the document from the server.
Blog Entries

- MBS Real Studio Plugins, version 12.4pr7

- MBS Real Studio Plugins, version 12.0pr5

18.1.39   IERunJavaScriptMBS(JavaScript as string) as boolean

Plugin Version: 8.0, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEWindowMBS class instead. **Function:** Runs a given javascript code.
Example:

```
// shows current user agent string which the control sends to website
call htmlviewer1.IERunJavaScriptMBS("document.title=navigator.userAgent;")
MsgBox htmlviewer1.IETitleMBS

// load a dummy page:
HTMLViewer1.LoadPage "<html><head></head><body></body></html>", nil

// shows an error message because of the navigator having a 6 inside:
call htmlviewer1.IERunJavaScriptMBS("document.title=navig6ator.userAgent;")

// uses try to avoid error message

dim script as string = "document.title=navig6ator.userAgent;"
dim s as string = "try { "+SCRIPT+" } catch(err) { document.title = err.description; } "

call htmlviewer1.IERunJavaScriptMBS(s)
MsgBox HTMLViewer1.IETitleMBS // shows "navig6ator is undefined"
```

Notes: Returns true if the javascript code was sent to the browser.
Returns false on any error.

The IE API does not allow to return values from Javascript. So you need to store your result in window.title and access it later using IETitleMBS.

IERunJavaScriptMBS fails if the htmlviewer is empty. You can load a dummy page like above.

You can use JavaScriptEngineMBS class to execute JavaScript without HTMLViewer in our own cross plat-form JavaScript engine.

See IEDocumentMBS.Eval for newer version which can return result.
Blog Entries

- HTMLViewer JavaScript communication for Xojo

- MBS Real Studio Plugins, version 12.1pr2

- Webviewer Tip

- Tipp of the day: Jump to anchor in htmlviewer

18.1.40   IEScrollHeightMBS as Integer

Plugin Version: 9.5, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Queries the height of the html viewer content.
Notes: Returns 0 on any error.
Blog Entries

- Picture from HTMLViewer

- MBS REALbasic plug-ins version 9.5

18.1.41   IEScrollWidthMBS as Integer

Plugin Version: 9.5, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Queries the width of the html viewer content.
Notes: Returns 0 on any error.
Blog Entries

- Picture from HTMLViewer

- MBS REALbasic plug-ins version 9.5

18.1.42   IESecurityMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves the security state.
Notes: Example value: "F√ºr diesen Dokumententyp gibt es kein Sicherheitszertifikat."

18.1.43 IESetTextAreaMBS(FormName as String, FieldName as String, Value as String) as Boolean

Plugin Version: 16.3, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets text for a textarea.
Notes: FormName can be ”” to look for any field with given name.
Raises exception if field is not found.
Returns true if text is set or false on failure.
Blog Entries

- HTMLViewer JavaScript communication for Xojo

- HTMLViewer JavaScript communication for Xojo

18.1.44 IEStopMBS as boolean

Plugin Version: 12.4, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEWebBrowserMBS class instead. **Function:** Stops loading.
Notes: Returns true on success and false on failure.
Blog Entries

- MBS Real Studio Plugins, version 12.4pr7

18.1.45 IETextMBS as string

Plugin Version: 7.7, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Returns a copy of the text of the current webpage.
Example:

msgbox htmlviewer1.IETextMBS

Notes: Asks Internet Explorer for a selection of the whole document and asks selection about the text content.
Returns ”” on any error.
Blog Entries

- MBS Xojo Plugins, version 20.2pr3

18.1.46 IEToStringMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Retrieves a string representation of the object.
Notes: Example value: " [object] "

18.1.47 IEWebBrowserMBS as IEWebBrowserMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries web browser object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.

Not available for DesktopHTMLViewer control since there is no support for Internet Explorer for that control.

18.1.48 IEWindowMBS as IEWindowMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries web window object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.

Not available for DesktopHTMLViewer control since there is no support for Internet Explorer for that control.

18.1.49 IEZoomMBS(factor as Integer) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEWebBrowserMBS class instead. **Function:** Zooms the web content.
Notes: Factor can be 50 for 50%.
Returns true on success.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.2pr1

18.1.50   Properties

18.1.51   IECharSetMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or retrieves the character set used to encode the object.
Notes: Example value: "utf-8"
(Read and Write computed property)

18.1.52   IECookieMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or gets the string value of a cookie.
Notes: Example value: "s_vi= [CS] v1 | 427CA13500002D10-A000B5B00000001 [CE] ; s_cc=true;
s_nr=1196708888562; s_sq=%5B%5BB%5D%5D"
(Read and Write computed property)

18.1.53   IEDefaultCharsetMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Gets the default character set from the current regional language settings.
Notes: Example value: "windows-1252"
(Read and Write computed property)

18.1.54   IEDomainMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or gets the security domain of the document.
Notes: Example value: "www.apple.com"
(Read and Write computed property)

18.1.55 IEEditableMBS as boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Makes the htmlviewer editable on Windows.
Notes: (Read and Write computed property)

18.1.56 IETitleMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or gets the title of the document.
Notes: Example value: "Apple"
(Read and Write computed property)

18.1.57 IEURLMBS as string

Plugin Version: 7.8, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. You can use methods in IEDocumentMBS class instead. **Function:** Sets or gets the URL for the current document.
Notes: Example value: "http://www.apple.com/"
(Read and Write computed property)

Chapter 19

HTMLViewer Win

19.1 class ChromiumBrowserMBS

19.1.1 class ChromiumBrowserMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The class for a WebKit browser on Windows.
Notes: Supports Xojo 2019r3 with MBS Plugin 20.1.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Upgrading WebKit for Windows support
- MBS Xojo Plugins, version 21.6pr3
- News from the MBS Xojo Plugins Version 21.3
- MBS Xojo Plugins, version 21.3pr1
- HTMLViewer JavaScript communication for Xojo
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- MBS Xojo Plugins, version 20.1pr6
- Upgrading our HTMLViewer functions for Internet Explorer
- HTMLViewer JavaScript communication for Xojo
- MBS Xojo / Real Studio plug-ins in version 14.2

Xojo Developer Magazine

- 18.3, page 9: News
- 12.4, page 8: News

19.1.2   Methods

19.1.3   AddCrossOriginWhitelist(SourceOrigin as String, TargetProtocol as String, TargetDomain as String, AllowTargetSubdomains as Boolean) as Boolean

Plugin Version: 21.3, Platform: Windows, Targets: Desktop only.

Function: Add an entry to the cross-origin access whitelist.
Example:

dim r as Boolean = ChromiumBrowserMBS.AddCrossOriginWhitelist("http://side1.mydomain.com", "http", "side2.mydomain.com", false)

Notes: The same-origin policy restricts how scripts hosted from different origins (scheme + domain + port) can communicate. By default, scripts can only access resources with the same origin. Scripts hosted on the HTTP and HTTPS schemes (but no other schemes) can use the "Access-Control-Allow-Origin" header to allow cross-origin requests. For example, https://source.example.com can make XMLHttpRequest requests on http://target.example.com if the http://target.example.com request returns an "Access-Control-Allow-Origin: https://source.example.com" response header.

Scripts in separate frames or iframes and hosted from the same protocol and domain suffix can execute cross-origin JavaScript if both pages set the document.domain value to the same domain suffix. For example, scheme://foo.example.com and scheme://bar.example.com can communicate using JavaScript if both domains set document.domain="example.com".

This function is used to allow access to origins that would otherwise violate the same-origin policy. Scripts hosted underneath the fully qualified SourceOrigin URL (like http://www.example.com) will be allowed access to all resources hosted on the specified TargetProtocol and TargetDomain.
If TargetDomain is non-empty and AllowTargetSubdomains if false only exact domain matches will be allowed. If TargetDomain contains a top-level domain component (like "example.com") and AllowTarget-Subdomains is true (1) sub-domain matches will be allowed. If TargetDomain is empty and AllowTarget-Subdomains if true (1) all domains and IP addresses will be allowed.

This function cannot be used to bypass the restrictions on local or display isolated schemes. See the comments on CefRegisterCustomScheme for more information.

This function may be called on any thread. Returns true on success or false if SourceOrigin is invalid or the whitelist cannot be accessed.

19.1.4   CanGoBack as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if the browser can navigate backwards.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.CanGoBack)

19.1.5 CanGoForward as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if the browser can navigate forwards.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.CanGoForward)

19.1.6 ClearCrossOriginWhitelist as Boolean

Plugin Version: 21.3, Platform: Windows, Targets: Desktop only.

Function: Remove all entries from the cross-origin access whitelist.
Example:

dim r as Boolean = ChromiumBrowserMBS.ClearCrossOriginWhitelist

Notes: Returns true on success or false if the whitelist cannot be accessed.

19.1.7 ClearFocus

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Clears focus.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.ClearFocus

19.1.8   ClearHistory

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Clear the back/forward browsing history.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.ClearHistory

Notes: Only supported for Chromium 2.x, but not 3.x.

19.1.9   CloseDevTools

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Explicitly close the developer tools window if one exists for this browser instance.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.CloseDevTools

19.1.10   Constructor

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

19.1.11   Destructor

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The destructor.

19.1.12 ExecuteJavaScript(jsCode as string, scriptUrl as string = ""), startLine as Integer = 0)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop only.

Function: Execute a string of JavaScript code in this frame.
Example:

// go back to last page via javascript
dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.ExecuteJavaScript "window.history.back();"

Notes: The scriptUrl parameter is the URL where the script in question can be found, if any. The renderer may request this URL to show the developer the source of the error. The startLine parameter is the base line number to use for error reporting.

19.1.13 Find(identifier as Integer, searchText as string, forward as boolean, MatchCase as boolean, FindNext as boolean)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Search for searchText.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.Find 1, "Hello", true, false, false

// later
m.Find 1, "Hello", true, false, true

Notes: Identifier can be used to have multiple searches running simultaniously. forward indicates whether to search forward or backward within the page. matchCase indicates whether the search should be case-sensitive. findNext indicates whether this is the first request or a follow-up.

19.1.14 Frame(ID as Int64) as ChromiumFrameMBS

Plugin Version: 17.2, Platform: Windows, Targets: Desktop only.

Function: Returns the frame with the specified identifier, or nil if not found.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
if b<>Nil then

```
dim idn() as Int64= b.FrameIdentifiers
dim ids() as string
for each n as Int64 in idn
dim f as ChromiumFrameMBS = b.Frame(n)

ids.Append str(n)+" "+f.Name
next
MsgBox "FrameIdentifiers: "+EndOfLine+EndOfLine+Join(ids, EndOfLine)
end if
```

See also:

19.1.15   Frame(name as string) as ChromiumFrameMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the frame with the specified name, or nil if not found.
Example:

```
dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.Frame("TopFrame")
dim s as string = f.Name
MsgBox s
```

See also:

19.1.16   FrameIdentifiers as Int64()

Plugin Version: 17.2, Platform: Windows, Targets: Desktop only.

Function: Queries list of frame identifiers.
Example:

```
dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
if b<>Nil then
dim idn() as Int64= b.FrameIdentifiers
dim ids() as string
for each n as Int64 in idn
dim f as ChromiumFrameMBS = b.Frame(n)
```

ids.Append str(n)+" "+f.Name
next
MsgBox "FrameIdentifiers: "+EndOfLine+EndOfLine+Join(ids, EndOfLine)
end if

19.1.17   FrameNames as String()

Plugin Version: 17.2, Platform: Windows, Targets: Desktop only.

Function: Queries list of frame names.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
if b<>Nil then
dim Names() as string = b.FrameNames
MsgBox "FrameNames: "+EndOfLine+EndOfLine+Join(Names, EndOfLine)
end if

Notes: Often frames are unnamed, so named is generic.

19.1.18   GoBack

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Navigate backwards.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.GoBack

19.1.19   GoForward

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Navigate forwards.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.GoForward

19.1.20   HidePopup

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Hide the currently visible popup, if any.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.HidePopup

Notes: Only supported for Chromium 2.x, but not 3.x.

19.1.21   Image(width as Integer, height as Integer) as Picture

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Get the raw image data contained in the specified element without performing validation.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim p as picture = m.Image(600, 800)
window1.backdrop = p

Notes: The specified width and height dimensions must match the current element size.

Only supported for Chromium 2.x, but not 3.x.

19.1.22   invalidate(x as Integer, y as Integer, width as Integer, height as Integer)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Invalidate the rectangle region of the view.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.invalidate 0, 0, 600, 800

Notes: This function is only used when window rendering is disabled and will result in a call to Handle-

Paint().

19.1.23 IsLoading as boolean

Plugin Version: 17.1, Platform: Windows, Targets: Desktop only.

Function: Checks loading state.
Notes: Returns true while page is loading.
Only available for Xojo 2017 and newer.

19.1.24 LibVersion as Integer

Plugin Version: 14.3, Platform: Windows, Targets: Desktop only.

Function: Returns revision number of the Chromium library.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(b.LibVersion)

Notes: up to Xojo 2014r1 this is 607.
Xojo 2014r2 uses 1562.

19.1.25 RegisterExtension(ExtensionName as String, javascriptCode as String) as Boolean

Plugin Version: 20.1, Platform: Windows, Targets: Desktop only.

Function: Register a new V8 extension with the specified JavaScript extension code.
Notes: Returns true on success.

19.1.26 Release

Plugin Version: 15.1, Platform: Windows, Targets: Desktop only.

Function: Releases the browser object.
Notes: You don't need this normally.
This method can be used for working around Feedback case 33565.

19.1.27 Reload

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Reload the current page.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.Reload

19.1.28 ReloadIgnoreCache

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Reload the current page ignoring any cached data.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.ReloadIgnoreCache

19.1.29 RemoveCrossOriginWhitelist(SourceOrigin as String, TargetProtocol as String, TargetDomain as String, AllowTargetSubdomains as Boolean) as Boolean

Plugin Version: 21.3, Platform: Windows, Targets: Desktop only.

Function: Remove an entry from the cross-origin access whitelist.
Example:

dim r as Boolean = ChromiumBrowserMBS.AddCrossOriginWhitelist("http://side1.mydomain.com", "http", "side2.mydomain.com", false)

Notes: Returns true on success or false if SourceOrigin is invalid or the whitelist cannot be accessed.

19.1.30 Retain

Plugin Version: 15.1, Platform: Windows, Targets: Desktop only.

Function: Retains the browser object.
Notes: You don't need this normally.

19.1.31 SetFocus(enableFocus as boolean = true)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Set focus for the browser window.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.SetFocus

Notes: If enable is true focus will be set to the window. Otherwise, focus will be removed.

19.1.32 setSize(width as Integer, height as Integer)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Set the size of the specified element.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.setSize 600, 800

Notes: Only supported for Chromium 2.x, but not 3.x.

19.1.33 ShowDevTools

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Open developer tools in its own window.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.ShowDevTools

19.1.34 StopFinding(clearSelection as boolean)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Cancel all searches that are currently going on.

19.1.35   StopLoad

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Stop loading the page.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
m.StopLoad

19.1.36   Properties

19.1.37   FocusedFrame as ChromiumFrameMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the focused frame for the browser window.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.FocusedFrame
dim s as string = f.URL
MsgBox s

Notes: (Read only property)

19.1.38   FrameCount as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop only.

Function: Queries frame count.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
if b<>Nil then
MsgBox "FrameCount: "+str(b.FrameCount)
end if

Notes: Only available in Xojo 2017 and newer.
(Read only property)

19.1.39 Handle as Integer

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

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

19.1.40 HasDocument as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if a document has been loaded in the browser.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.HasDocument)

Notes: (Read only property)

19.1.41 Height as Integer

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Queries height of the browser.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.Width) + " x " + str(m.Height)

Notes: Only supported for Chromium 2.x, but not 3.x.
(Read only property)

19.1.42 Identifier as Integer

Plugin Version: 20.1, Platform: Windows, Targets: Desktop only.

Function: Unique identifier.
Notes: (Read only property)

19.1.43   IsPopup as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if the window is a popup window.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.IsPopup)

Notes: (Read only property)

19.1.44   MainFrame as ChromiumFrameMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the main (top-level) frame for the browser window.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.MainFrame
dim s as string = f.URL
MsgBox s

Notes: (Read only property)

19.1.45   Parent as Variant

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The owner htmlviewer.
Notes: Can reference a Window or DesktopWindow object.
(Read only property)

19.1.46 PopupVisible as Boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if a popup is currently visible.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
msgbox str(m.PopupVisible)

Notes: Only supported for Chromium 2.x, but not 3.x.
(Read only property)

19.1.47 Width as Integer

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Queries width of the browser.
Example:

dim m as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
MsgBox str(m.Width) + " x " + str(m.Height)

Notes: Only supported for Chromium 2.x, but not 3.x.
(Read only property)

19.1.48 WindowRenderingDisabled as Boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if window rendering is disabled.
Notes: Only supported for Chromium 2.x, but not 3.x.
(Read only property)

19.1.49 ZoomLevel as Double

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Get/Set the zoom level.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.ZoomLevel = 2

Notes: Change the zoom level to the specified value.
(Read and Write property)

19.2   class ChromiumCookieManagerMBS

19.2.1   class ChromiumCookieManagerMBS

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The class to manage cookies for Chromium on Windows.
Notes: May need to have a HTMLViewer first being initialized by Xojo before the Chromium DLLs can work properly.

Supports Xojo 2019r3 with MBS Plugin 20.1.
Blog Entries

- MBS Xojo Plugins, version 23.1pr3

- Upgrading WebKit for Windows support

- Upgrading WebKit for Windows support

- MBS Xojo Plugins, version 20.1pr4

- Cookies in HTMLViewer

- MBS Releases the MBS Xojo / Real Studio plug-ins in version 15.2

- MBS Xojo / Real Studio Plugins, version 15.2pr2

- Cookies for Chromium in HTMLViewer on Windows

19.2.2   Methods

19.2.3   AllCookies as ChromiumCookieMBS()

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Visit all cookies.
Example:

dim m as new ChromiumCookieManagerMBS
dim cookies() as ChromiumCookieMBS = m.AllCookies
MsgBox str(cookies.Ubound+1)+" cookies"

Notes: The returned cookies are ordered by longest path, then by earliest creation date. Returns false (0) if cookies cannot be accessed.

19.2.4 Constructor

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The constructor.
Notes: Can raise exception if no cookie manager is available, e.g. when calling on Linux or Mac OS X or Windows if you don't have the dlls.
May not work if the Chromium was not initialized before by Xojo.
See also:

19.2.5 Constructor(path as string, PersistSessionCookies as Boolean)

Plugin Version: 16.4, Platform: Windows, Targets: Desktop only.

Function: The constructor for a new cookie manager.
Notes: Can raise exception if no cookie manager is available, e.g. when calling on Linux or Mac OS X or Windows if you don't have the dlls.
May not work if the Chromium was not initialized before by Xojo.
See also:

19.2.6 DeleteAllCookies as Integer

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Deletes all cookies.
Notes: Returns number of deleted cookies.

19.2.7 DeleteCookie(URL as string, CookieName as string) as boolean

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Delete all cookies that match the specified parameters.
Notes: If both url and values cookieName are specified all host and domain cookies matching both will be deleted. If only | url | is specified all host cookies (but not domain cookies) irrespective of path will be deleted. If url is empty all cookies for all hosts and domains will be deleted. Returns false if a non-empty invalid URL is specified or if cookies cannot be accessed.
Returns true on success.

19.2.8 DeleteCookies(URLs() as string, CookieNames() as string) as Integer

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Deletes several cookies.
Notes: Same as DeleteCookie, but with arrays for parameters.
Returns number of successful delete attempts.

19.2.9 DeleteURLCookies(URL as String, HTTPOnly as boolean = false) as Integer

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Deletes all cookies for a given URL.
Notes: The cookies to delete are filtered by the given url scheme, host, domain and path. If includeHttpOnly is true HTTP-only cookies will also be included in the deletion.
Returns number of cookies deleted.

19.2.10 Destructor

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The destructor.

19.2.11 SetCookie(URL as string, cookie as ChromiumCookieMBS) as boolean

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Sets a cookie given a valid URL and explicit user-provided cookie attributes.
Example:

dim d as new date
dim c as new ChromiumCookieMBS

c.Domain = "www.mbsplugins.de"
c.Path = "/"
c.CreationDate = d
c.LastAccessDate = d
c.ExpirationDate = nil
c.Value = "test "+d.SQLDateTime
c.Name = "test"
c.Secure = false
c.HTTPonly = false

if CookieManager.SetCookie(c.URL, c) then
msgbox "OK"
else
MsgBox "Failed to add cookie"
end if

Notes: This function expects each attribute to be well-formed. It will check for disallowed characters (e.g. the ';' character is disallowed within the cookie value attribute) and will return false without setting the cookie if such characters are found.

19.2.12   SetCookies(URL() as string, cookies() as ChromiumCookieMBS) as Integer

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Sets a lot of cookies.
Notes: Same as SetCookie, but with arrays of URLs and cookie objects.
Returns number of cookies created successfully.

19.2.13   SetStoragePath(Path as string) as boolean

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Sets the directory path that will be used for storing cookie data.
Example:

dim CookieManager as ChromiumCookieManagerMBS // property of window/app

dim f as FolderItem = SpecialFolder.Desktop.Child("cookies")
f.CreateAsFolder

CookieManager = new ChromiumCookieManagerMBS

if CookieManager.SetStoragePath(f.NativePath) then
MsgBox "OK"
else
MsgBox "Failed"
end if

Notes: If path is empty data will be stored in memory only.
Returns false if cookies cannot be accessed.

19.2.14   URLCookies(URL as String, HTTPOnly as boolean = false) as ChromiumCookieMBS()

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Queries a subset of cookies.
Notes: The results are filtered by the given url scheme, host, domain and path. If includeHttpOnly is true HTTP-only cookies will also be included in the results. The returned cookies are ordered by longest path, then by earliest creation date. Returns empty array if cookies cannot be accessed.
Returns nil on any error like low memory.

19.2.15   Properties

19.2.16   Handle as Integer

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The internal object refernce.
Notes: (Read only property)

19.3   class ChromiumCookieMBS

19.3.1   class ChromiumCookieMBS

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The class for a cookie with WebKit on Windows.
Notes: Supports Xojo 2019r3 with MBS Plugin 20.1.
Blog Entries

- Upgrading WebKit for Windows support

- Cookies in HTMLViewer

- MBS Releases the MBS Xojo / Real Studio plug-ins in version 15.2

- MBS Xojo / Real Studio Plugins, version 15.2pr2

- Cookies for Chromium in HTMLViewer on Windows

19.3.2   Methods

19.3.3   Constructor

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The constructor.

19.3.4   Destructor

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The destructor.

19.3.5   Properties

19.3.6   CreationDate as Date

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The cookie creation date.
Notes: This is automatically populated by the system on cookie creation.
(Read and Write property)

19.3.7 Domain as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The domain for this cookie.
Notes: If domain is empty a host cookie will be created instead of a domain cookie. Domain cookies are stored with a leading ”.” and are visible to sub-domains whereas host cookies are not.
(Read and Write property)

19.3.8 ExpirationDate as Date

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The cookie expiration date.
Notes: Can be nil to have no expiration date.
(Read and Write property)

19.3.9 HTTPonly as Boolean

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The HTTP only state.
Notes: If httponly is true the cookie will only be sent for HTTP requests.
(Read and Write property)

19.3.10 LastAccessDate as Date

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The cookie last access date.
Notes: This is automatically populated by the system on access.
(Read and Write property)

19.3.11 Name as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The cookie name.
Notes: (Read and Write property)

19.3.12   Path as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The path.
Notes: If path is non-empty only URLs at or below the path will get the cookie value.
(Read and Write property)

19.3.13   Scheme as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The scheme.
Notes: Depends on secure setting only.
Can be "http://" or "https://".
(Read only property)

19.3.14   Secure as Boolean

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: Whether this is a HTTPS only cookie.
Notes: If secure is true the cookie will only be sent for HTTPS requests.
(Read and Write property)

19.3.15   URL as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The URL of the cookie.
Notes: Build from Secure, Path and Domain.
(Read only property)

19.3.16   Value as String

Plugin Version: 15.2, Platform: Windows, Targets: Desktop only.

Function: The cookie value.
Notes: (Read and Write property)

19.4 class ChromiumFrameMBS

19.4.1 class ChromiumFrameMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The class for a webframe with WebKit on Windows.
Notes: Supports Xojo 2019r3 with MBS Plugin 20.1.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Upgrading WebKit for Windows support

- MBS Xojo Plugins, version 21.5pr1

- HTMLViewer JavaScript communication for Xojo

- HTMLViewer JavaScript communication for Xojo

- MBS Xojo Plugins, version 17.1pr4

- MBS Xojo / Real Studio plug-ins in version 14.2

- MBS Xojo / Real Studio Plugins, version 14.2pr7

19.4.2 Methods

19.4.3 Constructor

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

19.4.4 copy

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute copy in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.copy

19.4.5 cut

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute cut in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.cut

19.4.6 delete

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute delete in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.delete

19.4.7 Destructor

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The destructor.

19.4.8 ExecuteJavaScript(jsCode as string, scriptUrl as string = "", startLine as Integer = 0)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute a string of JavaScript code in this frame.
Example:

// go back to last page via javascript
dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
m.ExecuteJavaScript "window.history.back();"

Notes: The scriptUrl parameter is the URL where the script in question can be found, if any. The renderer may request this URL to show the developer the source of the error. The startLine parameter is the base

line number to use for error reporting.

You can use JavaScriptEngineMBS class to execute JavaScript without HTMLViewer in our own cross platform JavaScript engine.

19.4.9 LoadString(StringValue as string, URL as string)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Load the contents of StringValue with the optional dummy target URL.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
m.LoadString "<p>Hello</p>", "blank:about"

Notes: Since this was deprecated in Chromium and removed in newer function, it won't work for Xojo 2021r3 or newer. Please use data URL there or load html file URL.

19.4.10 LoadURL(URL as string)

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Load the specified url.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
m.LoadURL "http://www.macrumors.com"

19.4.11 paste

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute paste in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.paste

19.4.12   print

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Deprecated: This item is deprecated and should no longer be used. **Function:** Execute printing in the this frame. The user will be prompted with the print dialog appropriate to the operating system.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.print

Notes: Only supported for Chromium 2.x, but not 3.x.

19.4.13   redo

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute redo in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.redo

19.4.14   SelectAll

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute select all in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
m.SelectAll

19.4.15   undo

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Execute undo in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
b.MainFrame.undo

19.4.16   ViewSource

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Save this frame's HTML source to a temporary file and open it in the default text viewing application.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
m.ViewSource

19.4.17   Properties

19.4.18   Browser as ChromiumBrowserMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the browser that this frame belongs to.
Notes: (Read only property)

19.4.19   Handle as Integer

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

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

19.4.20   identifier as Int64

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the globally unique identifier for this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
MsgBox str(m.identifier)

Notes: (Read only property)

19.4.21   IsFocused as Boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if this is the focused frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
MsgBox str(m.IsFocused)

Notes: (Read only property)

19.4.22   IsMain as Boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns true if this is the main (top-level) frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
MsgBox str(m.IsMain)

Notes: (Read only property)

19.4.23   Name as String

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the name for this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim m as ChromiumFrameMBS = b.mainFrame
MsgBox m.Name

Notes: If the frame has an assigned name (for example, set via the iframe "name" attribute) then that value will be returned. Otherwise a unique name will be constructed based on the frame parent hierarchy. The main (top-level) frame will always have an empty name value.
(Read only property)

19.4.24 Parent as Variant

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: The owner htmlviewer.
Notes: Can reference a Window or DesktopWindow object.
(Read only property)

19.4.25 ParentFrame as ChromiumFrameMBS

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the parent of this frame or nil if this is the main (top-level) frame.
Notes: (Read only property)

19.4.26 Source as String

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns this frame's HTML source as a string.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.MainFrame
dim s as string = f.Source
Break // view in debugger

Notes: (Read only property)

19.4.27   Text as String

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns this frame's display text as a string.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.MainFrame
dim s as string = f.Text
Break // view in debugger

Notes: (Read only property)

19.4.28   URL as String

Plugin Version: 14.2, Platform: Windows, Targets: Desktop only.

Function: Returns the URL currently loaded in this frame.
Example:

dim b as ChromiumBrowserMBS = HTMLViewer1.ChromiumBrowserMBS
dim f as ChromiumFrameMBS = b.MainFrame
dim s as string = f.URL
MsgBox s

Notes: (Read only property)

19.5 class ChromiumWebPluginInfoMBS

19.5.1 class ChromiumWebPluginInfoMBS

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The class for details on installed plugins.
Notes: Supports Xojo 2019r3 with MBS Plugin 20.1.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Upgrading WebKit for Windows support

- Upgrading WebKit for Windows support

- MBS Xojo Plugins, version 20.1pr4

- MBS Xojo / Real Studio Plugins, version 16.0pr6

19.5.2 Methods

19.5.3 Constructor

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

19.5.4 Destructor

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The destructor.

19.5.5 Plugins as ChromiumWebPluginInfoMBS()

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: Queries list of plugins.
Notes: Those are the plugins Chromium found on the Windows PC.
e.g. 5 different versions of QuickTime Plugin.

19.5.6   Properties

19.5.7   Description as String

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: A description of the plugin from the version information.
Notes: (Read only property)

19.5.8   Name as String

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The plugin name (i.e. Flash).
Notes: (Read only property)

19.5.9   Path as String

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The plugin file path (DLL/bundle/library).
Notes: (Read only property)

19.5.10   Version as String

Plugin Version: 16.0, Platform: Windows, Targets: Desktop only.

Function: The version of the plugin (may be OS-specific).
Notes: (Read only property)

19.6 class IEDocumentMBS

19.6.1 class IEDocumentMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The class for the web document.
Notes: For HTMLViewer based on Internet Explorer on Windows.
All the methods and properties may raise IEExceptionMBS exceptions in case of failure.

Based on IHTMLDocument2 interface:
https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/aa752574(v=vs.8

Let us know if you miss something.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0

- MBS Xojo Plugins, version 21.0pr7

- MBS Xojo Plugins, version 20.5pr8

- HTMLViewer JavaScript communication for Xojo

- MBS Xojo Plugins, version 20.2pr3

- New in the MBS Xojo Plugins 20.0

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.0

- MBS Xojo Plugins, version 20.0pr6

- MBS Xojo Plugins, version 20.0pr5

- Upgrading our HTMLViewer functions for Internet Explorer

Xojo Developer Magazine

- 19.2, page 10: News

- 18.3, page 53: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

19.6.2 Methods

19.6.3 CallFunction(FunctionName as string, paramArray params as variant) as variant

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Calls a JavaScript function.
Example:

dim doc as IEDocumentMBS = htmlviewer1.IEDocumentMBS

dim v as variant = doc.CallFunctiom("test", 2, 3)

Notes: Returns result as variant.
You can pass as many parameters as needed.

Our plugin converts Xojo data types to JavaScript data types like boolean, number, text or NULL. This could be extended in future to convert arrays, too.

FunctionName should be the name of a global function defined in JavaScript.
See also:

19.6.4   CallFunction(FunctionName as string, params() as variant) as variant

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Calls a JavaScript function.
Example:

dim doc as IEDocumentMBS = htmlviewer1.IEDocumentMBS

dim Params() as Variant
Params.append 2
Params.append 3

dim v as variant = doc.CallFunctiom("test", Params)

Notes: Returns result as variant.
You can pass parameters for the function as array.

Our plugin converts Xojo data types to JavaScript data types like boolean, number, text or NULL. This could be extended in future to convert arrays, too.

FunctionName should be the name of a global function defined in JavaScript.
See also:

19.6.5 ClearBrowserSession as boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Ends browser session.
Notes: For HTMLViewer using Internet Explorer engine. If you close one by closing the window with HTMLViewer control, this method should end session and a new window would start fresh again.

Flushes entries not in use from the password cache on the hard disk drive. Also resets the cache time used when the synchronization mode is once-per-session.
This should clear current session cookies.

19.6.6 Constructor(DesktopHTMLViewer as DesktopHTMLViewer)

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

Function: Creates new web document object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.
See also:

- 19.6.7 Constructor(HTMLViewer as HTMLViewer) 535

19.6.7 Constructor(HTMLViewer as HTMLViewer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Creates new web document object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.
See also:

- 19.6.6 Constructor(DesktopHTMLViewer as DesktopHTMLViewer) 535

19.6.8 ContinueFindText(text as string, count as integer, flags as integer, selectText as boolean) as boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Continues a search started with IEFindTextMBS.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS

// find first

```
dim found as boolean = d.FindText("Xojo", 0, 0, true)

msgbox "Found Xojo: "+str(found)

if found then
// find next
found = d.ContinueFindText("Xojo", 0, 0, true)
end if
```

Notes: Parameters are the same as for IEFindTextMBS.

19.6.9   DrawToHDC(HDC as Ptr, PrinterName as string = "")

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Draws the content of the html document into the graphics context.
Notes: As of Windows Internet Explorer 9, this method is deprecated and should not be used.
Returns true on success and false on failure.

With some printers, running DrawToDC may cause problems. You can ensure that DrawToDC works properly on all printers by running SetDocumentPrinter method first, and then passing the modified device context to DrawToDC. The plugin calls SetDocumentPrinter for you when you provide a printer name.

19.6.10   Evaluate(expression as string) as variant

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Evaluates a JavaScript expression.
Example:

```
dim doc as IEDocumentMBS = htmlviewer1.IEDocumentMBS

dim v as variant = doc.Evaluate("1+2")
Msgbox v
```

Notes: In contrast to IEWindowMBS.RunJavaScript this function returns the result on Windows, but needs IE 9 or newer.

Our plugin converts Xojo data types to JavaScript data types like boolean, number, text or NULL. This could be extended in future to convert arrays, too.

19.6.11 FindText(text as string, count as integer, flags as integer, selectText as boolean) as boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Finds text on the current website.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS

// find first
dim found as boolean = d.FindText("Xojo", 0, 0, true)

msgbox "Found Xojo: "+str(found)

if found then
// find next
found = d.ContinueFindText("Xojo", 0, 0, true)
end if

Notes: text: the string that specifies the text to find.
count: long that specifies the number of characters to search from the starting point of the range. A positive integer indicates a forward search; a negative integer indicates a backward search.
Flags: integer that specifies one or more of the following flags to indicate the type of search:

0	Default. Match partial words.
1	Match backwards.
2	Match whole words only.
4	Match case.
131072	Match bytes.
536870912	Match diacritical marks.
1073741824	Match Kashida character.
2147483648	Match AlefHamza character.

Returns true: The search text was found.
Returns false: The search text was not found.

19.6.12 Frames as IEWindowMBS()

Plugin Version: 21.0, Platform: Windows, Targets: Desktop only.

Function: Queries the collection of frames.

Example:

Dim hdoc As IEDocumentMBS = HTMLViewer1.IEDocumentMBS
Dim frames() As IEWindowMBS = hdoc.Frames

For Each frame As IEWindowMBS In frames
Dim doc As IEDocumentMBS = frame.Document

Dim URL As String = doc.URL
Dim Name As String = doc.NameProp
Dim HTMLText As String = doc.HTMLText

Break // read in debugger
Next

Notes: MBS Plugin will return the window objects for each frame as an array.

19.6.13   GetTextArea(FormName as String, FieldName as String) as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries text for a textarea.
Notes: FormName can be ”” to look for any field with given name.
Raises exception if field is not found.
Returns text from textarea.

19.6.14   HTMLText as string

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Returns a copy of the html source code of the current webpage.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim htmlText as string = d.HTMLText
dim PlainText as string = d.text
dim toString as string = d.ToString

Break // see in debug

Notes: Improved in plugin version 12.2 to return better HTML text. This is the html generated from current web content and not the page we originally loaded. So this works with Editable property.

Returns "" on any error.

19.6.15 Image as picture

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Copies the picture from the document.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim p as Picture = d.Image
canvas1.Backdrop = p

Notes: You may want to resize the htmlviewer to get a picture without scrollbars. (See example projects)
You may need to call ClearFocus as it seems like if the focus is on the htmlviewer it does not draw itself in our picture.

19.6.16 LoadHTML(HTMLText as string)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Loads the HTML text into the htmlviewer.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS

d.LoadHTML "<p>Hello World</p>"

Notes: Does not use a temp file like Xojo's built in method.
Returns true on success.
On Windows you may need to reset webviewer before or load "about:blank" to initialize the webviewer by Xojo (or Xojo).

19.6.17 PrintPreview as boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Commands Internet Explorer to show the print preview dialog for this htmlviewer.
Notes: Returns true on success. Returns false if function is not supported.
The function returns directly while the preview dialog is still running.

19.6.18   Reload(Force as boolean = false)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Reloads the current page.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
d.Reload

Notes: Boolean that specifies one of the following possible values:
False: Default. Reloads the document from the cache.
True: Reloads the document from the server.

19.6.19   SetTextArea(FormName as String, FieldName as String, Value as String) as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets text for a textarea.
Notes: FormName can be ”” to look for any field with given name.
Raises exception if field is not found.
Returns true if text is set or false on failure.

19.6.20   Text as string

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Returns a copy of the text of the current webpage.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim htmlText as string = d.HTMLText
dim PlainText as string = d.text
dim toString as string = d.ToString

Break // see in debug

Notes: Asks Internet Explorer for a selection of the whole document and asks selection about the text
content.
Returns ”” on any error.

19.6.21 Properties

19.6.22 CharSet as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or retrieves the character set used to encode the object.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.CharSet

Notes: Example value: "utf-8"
(Read and Write property)

19.6.23 Cookie as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets the string value of a cookie.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.Cookie

Notes: Example value: "s_vi= [CS] v1 | 427CA13500002D10-A000B5B00000001 [CE] ; s_cc=true; s_nr=1196708888562; s_sq=%5B%5BB%5D%5D"
(Read and Write property)

19.6.24 DefaultCharset as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the default character set from the current regional language settings.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.DefaultCharset

Notes: Example value: "windows-1252"
(Read and Write property)

19.6.25   Domain as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets the security domain of the document.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.Domain

Notes: Example value: "www.apple.com"
(Read and Write property)

19.6.26   Editable as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Whether the htmlviewer is editable on Windows.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
d.Editable = true

Notes: (Read and Write property)

19.6.27   FileCreationDate as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the date the file was created.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.FileCreationDate

Notes: Example value: "09/13/2007"
(Read only property)

19.6.28 FileModifiedDate as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the date the file was last modified.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.FileModifiedDate

Notes: Example value: "12/03/2007"
(Read only property)

19.6.29 FileSize as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the file size.
Notes: Example value: "12475"
(Read only property)

19.6.30 FileUpdatedDate as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the date the file was last updated.
Notes: Example value: "01/01/1601"
(Read only property)

19.6.31 Handle as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

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

19.6.32 History as IEHistoryMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries history object for web document.
Notes: (Read only property)

19.6.33   LastModified as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the date that the page was last modified, if the page supplies one.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.LastModified

Notes: Example value: ”12/03/2007 20:08:17”
(Read only property)

19.6.34   MimeType as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the MIME type for the file.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.MimeType

Notes: Example value: ”HTM-Datei”
See also MimeTypeToFileExtensionMBS function.
(Read only property)

19.6.35   NameProp as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the title of the document file.
Notes: Example value: ”Apple”
(Read only property)

19.6.36 Navigator as IENavigatorMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries navigator object for web document.
Notes: (Read only property)

19.6.37 ParentWindow as IEWindowMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries parent window for this document.
Notes: (Read only property)

19.6.38 Protocol as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or retrieves the protocol portion of a URL.
Notes: Example value: "HTTP (HyperText Transfer-Protokoll)"
(Read only property)

19.6.39 ReadyState as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves a value that indicates the current state of the htmlviewer.
Notes: uninitialized: Object is not initialized with data.
loading: Object is loading its data.
loaded: Object has finished loading its data.
interactive: User can interact with the object even though it is not fully loaded.
complete: Object is completely initialized.

An object's state is initially set to uninitialized, and then to loading. When data loading is complete, the state of the link object passes through the loaded and interactive states to reach the complete state.

The states through which an object passes are determined by that object; an object can skip certain states (for example, interactive) if the state does not apply to that object.

Data source objects and databound elements are normally populated asynchronously, and certain programmatic operations can only be performed reliably on databound objects when they are ready for use. There-

fore, the appropriate code should be written to confirm the readyState of objects prior to performing certain operations on them. For example, walking the rows of a table should not be attempted until after the table has reached the complete state.

The readyState property enables the status of an object to be tested. The correct place to test the readyState property is in the event handler for onreadystatechange. Similarly, a data source object (DSO) fires the ondatasetcomplete event to notify the document that the dataset is ready for programmatic operation. (Read only property)

19.6.40 Referrer as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the URL of the location that referred the user to the current page.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.Referrer

Notes: Example value: "http://www.apple.com/"
(Read only property)

19.6.41 ScrollHeight as integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries the height of the html viewer content.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS

msgbox str(d.ScrollWidth)+" "+str(d.ScrollHeight)

Notes: Returns 0 on any error.
(Read only property)

19.6.42 ScrollWidth as integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries the width of the html viewer content.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS

msgbox str(d.ScrollWidth)+" "+str(d.ScrollHeight)

Notes: Returns 0 on any error.
(Read only property)

19.6.43   Security as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the security state.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.Security

Notes: Example value: "F√°r diesen Dokumententyp gibt es kein Sicherheitszertifikat."
(Read only property)

19.6.44   Title as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets the title of the document.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.Title

Notes: Example value: "Apple"
(Read and Write property)

19.6.45   ToString as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves a string representation of the object.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim htmlText as string = d.HTMLText
dim PlainText as string = d.text
dim toString as string = d.ToString

Break // see in debug

Notes: Example value: " [object] "
(Read only property)

19.6.46   URL as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets the URL for the current document.
Example:

dim d as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
msgbox d.URL

Notes: Example value: "http://www.apple.com/"
(Read and Write property)

19.7 class IEExceptionMBS

19.7.1 class IEExceptionMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for an error in one of the IE classes.
Notes: See errorNumber and message properties.
Subclass of the RuntimeException class.
Blog Entries

- Upgrading our HTMLViewer functions for Internet Explorer

19.8   class IEHistoryMBS

19.8.1   class IEHistoryMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The class for the web history.
Notes: For HTMLViewer based on Internet Explorer on Windows.
All the methods and properties may raise IEExceptionMBS exceptions in case of failure.

Based on IOmHistory interface:
https://docs.microsoft.com/en-us/previous-versions/aa703740(v=vs.85)

Let us know if you miss something.
Blog Entries

- MBS Xojo Plugins, version 20.5pr9

- MBS Xojo Plugins, version 20.0pr5

- Upgrading our HTMLViewer functions for Internet Explorer

19.8.2   Methods

19.8.3   Back

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Loads a previous URL from the History list.

19.8.4   Constructor(DesktopHTMLViewer as DesktopHTMLViewer)

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

Function: Constructor to get the history for a HTMLViewer easily.
See also:

19.8.5   Constructor(HTMLViewer as HTMLViewer)

Plugin Version: 20.5, Platform: Windows, Targets: Desktop only.

Function: Constructor to get the history for a HTMLViewer easily.
See also:

19.8.6   Forward

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Loads the next URL from the History list.

19.8.7   Go(index as Integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Loads a URL from the History list.
Notes: index: An integer indicates the relative position of a URL in the History list.

An error does not occur if the user tries to go beyond the beginning or end of the history. Instead, the user remains at the current page.

19.8.8   Properties

19.8.9   Handle as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

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

19.8.10   Length as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the number of elements in the History list.
Notes: Example value: 0
(Read only property)

19.9   class IENavigatorMBS

19.9.1   class IENavigatorMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The class for a navigator object.
Notes: For HTMLViewer based on Internet Explorer on Windows.
All the methods and properties may raise IEExceptionMBS exceptions in case of failure.

Based on IOmNavigator interface:
https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/hh774262(v=vs.8

Let us know if you miss something.
Blog Entries

- MBS Xojo Plugins, version 20.5pr9

- MBS Xojo Plugins, version 20.0pr5

- Upgrading our HTMLViewer functions for Internet Explorer

19.9.2   Methods

19.9.3   Constructor(DesktopHTMLViewer as DesktopHTMLViewer)

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

Function: Constructor to get the navigator for a HTMLViewer easily.
See also:

- 19.9.4 Constructor(HTMLViewer as HTMLViewer)   552

19.9.4   Constructor(HTMLViewer as HTMLViewer)

Plugin Version: 20.5, Platform: Windows, Targets: Desktop only.

Function: Constructor to get the navigator for a HTMLViewer easily.
See also:

- 19.9.3 Constructor(DesktopHTMLViewer as DesktopHTMLViewer)   552

19.9.5 Properties

19.9.6 AppCodeName as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the code name.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.AppCodeName

Notes: Returns "Mozilla". Returned by Internet Explorer and Netscape Navigator.
(Read only property)

19.9.7 AppMinorVersion as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the application's minor version value.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.AppMinorVersion

Notes: Example value: ";SP2;"
e.g. "0"
(Read only property)

19.9.8 AppName as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the name of the browser.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.AppName

Notes: Example value: "Microsoft Internet Explorer"
(Read only property)

19.9.9   AppVersion as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the platform and version of the browser.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.AppVersion

Notes: Example value: "4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 2.0.50727)"
(Read only property)

19.9.10   BrowserLanguage as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the current browser language.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.BrowserLanguage

Notes: Example value: "de"
(Read only property)

19.9.11   CookieEnabled as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves whether client-side persistent cookies are enabled in the browser.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

if a.CookieEnabled then
msgbox "Cookie enabled"
else
msgbox "Cookie disabled"
end if

Notes: Persistent cookies are those that are stored on the client-side computer.
Example value: True
(Read only property)

19.9.12   CPUClass as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves a string denoting the CPU class.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.CPUClass

Notes: Returns "x86" for Intel processor.
(Read only property)

19.9.13   Handle as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

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

19.9.14   JavaEnabled as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Returns whether Java is enabled.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

if a.JavaEnabled then
msgbox "Java enabled"
else
msgbox "Java disabled"
end if

Notes: Example value: True
(Read only property)

19.9.15   OnLine as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves a value indicating whether the system is in global offline mode.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.OnLine

Notes: The user can modify the global offline state by choosing Work Offline from the File menu in Microsoft Internet Explorer version 4.0 or later. This property does not indicate whether the system is connected to the network.

Example value: True
(Read only property)

19.9.16   Platform as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the name of the user's operating system.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.Platform

Notes: Returns "Win64" for Windows 64-bit platform.
Returns "Win32" for Windows 32-bit platform.
(Read only property)

19.9.17   SystemLanguage as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the default language used by the operating system.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.SystemLanguage

msgbox a.AppCodeName

Notes: The systemLanguage property reflects the language edition of the installed operating system.
e.g. de-DE
(Read only property)

19.9.18   UserAgent as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves a string equivalent to the HTTP user-agent request header.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.UserAgent

Notes: The HTTP user-agent request header contains information about compatibility, the browser, and

the platform name. For more information about the browser, see the IENavigatorappNameMBS property. For more information about the platform, see the IENavigatorappVersionMBS property.

The IENavigatoruserAgentMBS property dynamically returns a different value depending on the browser and platform versions. For example, Microsoft Internet Explorer 6 returns the following string for Microsoft Windows XP.

Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)

Example value: "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 2.0.50727)"

Like other plugin function, this can only work if a page has been loaded. Xojo won't create the internal htmlviewer object before you load a page.
(Read only property)

19.9.19   UserLanguage as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the operating system's natural language setting.
Example:

dim w as IEDocumentMBS = HTMLViewer1.IEDocumentMBS
dim a as IENavigatorMBS = w.Navigator

msgbox a.AppCodeName

Notes: Example value: "de-DE"
(Read only property)

19.10   class IEWebBrowserMBS

19.10.1   class IEWebBrowserMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The class for a web browser.
Notes: For HTMLViewer based on Internet Explorer on Windows.
All the methods and properties may raise IEExceptionMBS exceptions in case of failure.

Based on IWebBrowser2 interface:
https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/aa752127(v=vs.8

Let us know if you miss something.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3

- MBS Xojo Plugins, version 20.5pr9

- News from the MBS Xojo Plugins Version 20.3

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.3

- Using DebugBar in Xojo for IE HTMLViewer

- HTMLViewer JavaScript communication for Xojo

- News from the MBS Xojo Plugins Version 20.1

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1

- New in the MBS Xojo Plugins 20.0

- Upgrading our HTMLViewer functions for Internet Explorer

Xojo Developer Magazine

- 18.3, page 53: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

- 18.3, page 10: News

19.10.2   Methods

19.10.3   Constructor(DesktopHTMLViewer as DesktopHTMLViewer)

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

Function: Constructor to get the browser for a HTMLViewer easily.
See also:

19.10.4   Constructor(HTMLViewer as HTMLViewer)

Plugin Version: 20.5, Platform: Windows, Targets: Desktop only.

Function: Constructor to get the browser for a HTMLViewer easily.
See also:

19.10.5   GetInternetExplorerHiDPI as Integer

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: Query whether HiDPI handling is enabled.
Notes: We can opt-in for your application to let HTMLviewer use proper resolution handling for web content.

19.10.6   GetInternetExplorerVersion as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries IE version used for this application.
Notes: Queries the Internet Explorer version to use for HTMLViewer.

19.10.7   GoBack

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Navigates backward one item in the history list.

19.10.8   GoForward

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Navigates forward one item in the history list.

19.10.9 GoHome

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Navigates to the current home or start page.
Notes: The user can indicate the URL to use for the home page from the Internet Options dialog box, which is accessible from the Tools menu of Windows Internet Explorer, or in the Control Panel.
Internet Explorer 7 and later. This method navigates to the first URL in the home tabs group.

19.10.10 GoSearch

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Navigates to the current search page.
Notes: The user can indicate the URL to use for the search page with the Internet Options dialog box, which is accessible from the Tools menu of Windows Internet Explorer, or in the Control Panel.
Internet Explorer 7 and later. GoSearch navigates to the default search provider. For more information, see Windows Search Guide.

19.10.11 Navigate(URL as string, Flags as Integer = 0, TargetFrameName as String = "", PostData as String = "", Headers as String = "")

Plugin Version: 20.1, Platform: Windows, Targets: Desktop only.

Function: Navigates to a resource identified by a URL or to a file identified by a full path.
Notes: URL: The URL, full path, or Universal Naming Convention (UNC) location and name of the resource to display.
Flags: The combination of Navigate* constants.
TargetFrameName: Optional, the name of the frame in which to display the resource.
PostData: The post data that is sent to the server as part of a HTTP POST transaction. A POST transaction is typically used to send data gathered by an HTML form. If this parameter does not specify any post data, this method issues an HTTP GET transaction. This parameter is ignored if the URL is not an HTTP URL.
Headers: The text list that contains additional HTTP headers to send to the server. These headers are added to the default headers. For example, headers can specify the action required of the server, the type of data being passed to the server, or a status code. This parameter is ignored if the URL is not an HTTP URL.

The possible values for the TargetFrameName parameter are:

19.10.12 Refresh

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

_blank	Load the link into a new unnamed window.
_parent	Load the link into the immediate parent of the document the link is in.
_self	Load the link into the same window the link was clicked in.
_top	Load the link into the full body of the current window.
WindowName	A named HTML frame. If no frame or window exists that matches the specified target name, a new window is opened for the specified link.

Function: Reloads the file that is currently displayed in the object.
Notes: This method is the same as clicking the Refresh button or pressing F5 in Windows Internet Explorer. To save time and network bandwidth, the WebBrowser control and InternetExplorer application store pages from recently visited sites in cached memory on the user's hard disk. When you revisit a Web site, the page is reloaded from the local disk instead of being downloaded again from the server. To ensure that you are viewing the most current version of the page, you can force a fresh download by using the Refresh method. This method adds the "Pragma: No-cache" HTTP header to the request.
See also:

19.10.13   Refresh(Level as Integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Reloads the file that is currently displayed with the specified refresh level.
See also:

19.10.14   SetInternetExplorerHiDPI(Enable as Boolean) as Boolean

Plugin Version: 20.2, Platform: Windows, Targets: Desktop only.

Function: Sets whether HiDPI handling is enabled.
Notes: We can opt-in for your application to let HTMLviewer use proper resolution handling for web content.
You may just call this in your app.open event.

19.10.15   SetInternetExplorerVersion(version as Integer) as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets IE version used for this application.
Example:

// make sure we don't end up with Internet Explorer in version 7, which is default.
Dim v As Integer = IEWebBrowserMBS.GetInternetExplorerVersion

If v <11000 Then // below version 11.0

Dim b As Boolean = IEWebBrowserMBS.SetInternetExplorerVersion(11000)
If b Then
MsgBox "Requested Internet Explorer version 11.0."
Else
MsgBox "Failed to set IE version?"
End If
End If

Notes: By default Xojo uses IE7 for the HTMLViewer, even if you have version 11 enabled.

In a test on Windows 10, we get by default this browser version:

Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.2; Win64; x64; Trident/7.0; .NET4.0C; .NET4.0E; Tablet PC 2.0; .NET CLR 2.0.50727; .NET CLR 3.0.30729; .NET CLR 3.5.30729)

and with setting it to 11000, we get this:

Mozilla/5.0 (Windows NT 6.2; Win64; x64; Trident/7.0; rv:11.0) like Gecko

19.10.16 ShowDebugBar(NoScale as Boolean = false, x as Integer = 0, y as Integer = 0, width as Integer = 0, Height as Integer = 0, TopMost as Boolean = false) as Boolean

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: Shows the DebugBar for the web viewer on Windows.
Notes: With DebugBar you can inspect HTML elements, see scripts and modify elements on the fly. There is a JavaScript console to quickly edit something if needed.

Please download and install DebugBar extension for Internet Explorer:
https://www.debugbar.com

You may need to buy a license for it to use all features.

Known issues:

- The toolbar has a gear button, where you can pick locations. Choosing a location there may crash the application.

- If application runs on a screen with >100% scaling, the picker may not work correctly unless you pass 1 for Flags parameter, but then icons may be smaller.

- HTTPs tab shows no data.

- You may need to click reload in the DebugBar (orange arrow in toolbar) when switching to another page.

- Running JavaScript works, but doesn't show errors or results.

- The inspect element in context menu command does not work.

Returns true on success or false on failure.

NoScale: disable scaling on >100% screen.
TopMost: Put the window always on top of other windows.
X, Y, width and height allow to position the window. Default is near center of screen with 600x400 in size.

19.10.17   Stop

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Stops loading.
Notes: Returns true on success and false on failure.

19.10.18   Zoom(factor as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Zooms the web content.
Notes: Factor can be 50 for 50%.
Returns true on success.

19.10.19   Properties

19.10.20   AddressBar as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the address bar of the object is visible or hidden.
Notes: (Read and Write property)

19.10.21   Busy as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets a value that indicates whether the object is engaged in a navigation or downloading operation.
Notes: If the control is busy, you can use the Stop method to cancel the navigation or download operation before it is completed.
(Read only property)

19.10.22   FullName as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the fully qualified path of the Windows Internet Explorer executable.
Notes: Internet Explorer 8 and later. Always returns an empty string.
This method also updates the window list.
(Read only property)

19.10.23   FullScreen as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether Windows Internet Explorer is in full-screen mode or normal window mode.
Notes: True: Internet Explorer is in full-screen mode.
False: Default. Internet Explorer is in normal window mode.

In full-screen mode, the Internet Explorer main window is maximized, and the status bar, toolbar, menu bar, and title bar are hidden.
Setting FullScreen (even to false) resets the values of the AddressBar and ToolBar properties to true. Disable the address bar and toolbars after you set the FullScreen property.
The WebBrowser object saves the value of this property, but otherwise ignores it.
(Read and Write property)

19.10.24   Handle as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

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

19.10.25   LocationName as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the path or title of the resource that is currently displayed.
Notes: If the resource is an HTML webpage, the name is the title of the document. If the resource is a folder or file on the network or local computer, the name is the full file system path of the folder or file in Universal Naming Convention (UNC) format.
If this method is called before the first navigation is complete, InternetExplorer returns an empty string.
(Read only property)

19.10.26   LocationURL as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the URL of the resource that is currently displayed.
Example:

dim w as IEWebBrowserMBS = HTMLViewer1.IEWebBrowserMBS
msgbox w.LocationURL

Notes: If the resource is a folder or file on the network or local computer, the name is the full path of the folder or file in the Universal Naming Convention (UNC) format.
If this method is called before the first navigation is complete, InternetExplorer returns an empty string.
(Read only property)

19.10.27   MenuBar as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the Windows Internet Explorer menu bar is visible.
Notes: True: Default. Menu bar is visible.
False: Menu bar is hidden.
(Read and Write property)

19.10.28   Name as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The frame name or application name of the object.
Example:

dim w as IEWebBrowserMBS = HTMLViewer1.IEWebBrowserMBS
msgbox w.Name

Notes: (Read only property)

19.10.29   Offline as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the object is operating in offline mode.
Notes: In offline mode, the browser is forced to read HTML pages from the local cache instead of reading from the source document online.
The WebBrowser object delegates this method to the top-level frame. If no frame exists, it returns false.
(Read and Write property)

19.10.30   ReadyState as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the ready state of the object.
Notes: (Read only property)

19.10.31   RegisterAsBrowser as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the object is registered as a top-level browser window.
Notes: The architecture of Windows Internet Explorer allows for multiple instances of the IWebBrowser2 object. A "top-level" IWebBrowser2 object is the parent of all other IWebBrowser2 objects. Only top-level objects can register as a browser with this method.
By setting this property to True, Internet Explorer and Windows can locate other top-level IWebBrowser2 objects that can participate in frame name resolution. For example, if the user clicks on a hyperlink that specifies a target attribute that is not in the current window, Internet Explorer searches all registered IWeb-Browser2 objects in an attempt to resolve the target name.
After the window is registered as a top-level browser, it cannot be unregistered.

(Read and Write property)

19.10.32 RegisterAsDropTarget as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the object is registered as a drop target for navigation.
Notes: A drop target is an object that has been registered to accept data from a drag-and-drop operation.
The WebBrowser object delegates this method to the top-level frame. If no frame exists, it raises exception.
(Read and Write property)

19.10.33 Silent as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the object can display dialog boxes.
Notes: False: Default. Dialog boxes and messages can be displayed. Critical errors and security alerts are not suppressed.
True: Dialog boxes are not displayed.

The WebBrowser object delegates this method to the top-level frame. If no frame exists, it raises exception.
(Read and Write property)

19.10.34 StatusBar as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets a value that indicates whether the status bar for the object is visible.
Notes: The WebBrowser object saves the value of this property, but otherwise ignores it.
(Read and Write property)

19.10.35 StatusText as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets the text in the status bar for the object.
Example:

dim w as IEWebBrowserMBS = HTMLViewer1.IEWebBrowserMBS
msgbox w.StatusText

Notes: Windows Internet Explorer 7 and later. The security settings that restrict access to the status bar from script do not apply to this method.
(Read and Write property)

19.10.36   TheaterMode as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets whether the object is in theater mode.
Notes: True: Object is in theater mode.
False: Default. Object is in normal mode.

In theater mode, the object's main window fills the entire screen and displays a toolbar that has a minimal set of navigational buttons. A status bar is also provided in the upper-right corner of the screen. Explorer bars, such as History and Favorites , are displayed as an autohide pane on the left edge of the screen in theater mode.

Setting TheaterMode (even to false) resets the values of the AddressBar and ToolBar properties to true. Disable the address bar and toolbars after you set the TheaterMode property.
The WebBrowser object saves the value of this property, but otherwise ignores it.
(Read and Write property)

19.10.37   ToolBar as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or gets whether toolbars for the object are visible.
Notes: False: Toolbar is hidden.
True: Default. Toolbar is visible.

When the ToolBar property is set to falsde, it is not equivalent to the "toolbar=no" feature of window.open. Instead, it turns off all user interface elements that can be considered toolbars, leaving Windows Internet Explorer in a blank state.
The WebBrowser object saves the value of this property, but otherwise ignores it.
(Read and Write property)

19.10.38   Type as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Gets the user type name of the contained document object.
Notes: For example, this method returns "HTML Document" if the document supports the IHTMLDocument2 interface.

The WebBrowser object delegates this method to the top-level frame. If no frame exists, it returns empty text.
(Read only property)

19.10.39   Constants

Constants

Constant	Value	Description
RefreshIfExpired	1	One of the Refresh levels.
		Not currently implemented.
RefreshNormal	0	One of the Refresh levels.
		Refresh without sending a "Pragma:no-cache" HTTP header to the server.

Navigation Flags

Constant	Value	Description
NavigateAllowAutosearch	&h10	If the navigation fails, the autosearch functionality attempts to navigate common root domains (.com, .edu, and so on). If this also fails, the URL is pa to a search engine.
NavigateBlockRedirectsXDomain	&h8000	Internet Explorer 8. Block cross-domain redirect requests.
NavigateBrowserBar	&h20	Causes the current Explorer Bar to navigate to the given item, if possible
NavigateEnforceRestricted	&h80	Internet Explorer 6 for Windows XP SP2 and later. Force the URL into restricted zone.
NavigateHyperlink	&h40	Internet Explorer 6 for Windows XP SP2 and later. If the navigation fails a hyperlink is being followed, this constant specifies that the resource sh then be bound to the moniker using the BINDF_HYPERLINK flag.
NavigateKeepWordWheelText	&h2000	Internet Explorer 7. Maintain state for dynamic navigation based on the string entered in the search band text box (wordwheel). Restore the wordw text when the navigation completes.
NavigateNewWindowsManaged	&h100	Internet Explorer 6 for Windows XP SP2 and later. Use the default Po Manager to block pop-up windows.
NavigateNoHistory	2	Do not add the resource or file to the history list. The new page replaces current page in the list.
NavigateNoReadFromCache	4	Not implemented.
NavigateNoWriteToCache	8	Not implemented.
NavigateOpenInBackgroundTab	&h1000	Internet Explorer 7. Open the resource or file in a new background tab; currently active window and/or tab remains open on top.
NavigateOpenInNewTab	&h800	Internet Explorer 7. Open the resource or file in a new tab. Allow the d nation window to come to the foreground, if necessary.
NavigateOpenNewForegroundTab	&h10000	Internet Explorer 8 and later. Open the resource in a new tab that beco the foreground tab.
NavigateTrustedForActiveX	&h400	Internet Explorer 6 for Windows XP SP2 and later. Prompt for the installa of ActiveX controls.
NavigateUntrustedForDownload	&h200	Internet Explorer 6 for Windows XP SP2 and later. Block files that norn trigger a file download dialog box.
NavigateVirtualTab	&h4000	Internet Explorer 8. Open the resource as a replacement for the curren target tab. The existing tab is closed while the new tab takes its place in the bar and replaces it in the tab group, if any. Browser history is copied forv to the new tab. On Windows Vista, this flag is implied if the navigation w cross integrity levels and navOpenInNewTab, navOpenInBackgroundTab navOpenInNewWindow is not specified.

Ready States

Constant	Value	Description
ReadyStateComplete	4	Object has received all of its data.
ReadyStateInteractive	3	Object is interactive, but not all of its data is available.
ReadyStateLoaded	2	Object has been initialized.
ReadyStateLoading	1	Object is currently loading its properties.
ReadyStateUninitialized	0	Default initialization state.

Refresh Levels

Constant	Value	Description
RefreshCompletely	2	Refresh without forced cache validation by sending a "Pragma:no-cache" header to the server (HTTP URLs only). Same as pressing Ctrl+F5 in Microsoft Internet Explorer.

Value	Description
11001	Internet Explorer 11. Webpages are displayed in IE11 edge mode, regardless of the declared !DOCTYPE directive. Failing to declare a !DOCTYPE directive causes the page to load in Quirks.
11000	IE11. Webpages containing standards-based !DOCTYPE directives are displayed in IE11 edge mode. Default value for IE11.
10001	Internet Explorer 10. Webpages are displayed in IE10 Standards mode, regardless of the !DOCTYPE directive.
10000	Internet Explorer 10. Webpages containing standards-based !DOCTYPE directives are displayed in IE10 Standards mode. Default value for Internet Explorer 10.
9999	Windows Internet Explorer 9. Webpages are displayed in IE9 Standards mode, regardless of the declared !DOCTYPE directive. Failing to declare a !DOCTYPE directive causes the page to load in Quirks.
9000	Internet Explorer 9. Webpages containing standards-based !DOCTYPE directives are displayed in IE9 mode. Default value for Internet Explorer 9. In Internet Explorer 10, Webpages containing standards-based !DOCTYPE directives are displayed in IE10 Standards mode.
8888	Webpages are displayed in IE8 Standards mode, regardless of the declared !DOCTYPE directive. Failing to declare a !DOCTYPE directive causes the page to load in Quirks.
8000	Webpages containing standards-based !DOCTYPE directives are displayed in IE8 mode. Default value for Internet Explorer 8. In Internet Explorer 10, Webpages containing standards-based !DOCTYPE directives are displayed in IE10 Standards mode.
7000	Webpages containing standards-based !DOCTYPE directives are displayed in IE7 Standards mode. Default value for applications hosting the WebBrowser Control.

19.11   class IEWindowMBS

19.11.1   class IEWindowMBS

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: The class for a web window.
Notes: For HTMLViewer based on Internet Explorer on Windows.
All the methods and properties may raise IEExceptionMBS exceptions in case of failure.

Based on IHTMLWindow2 interface:
https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/aa741505(v=vs.8

Let us know if you miss something.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0

- MBS Xojo Plugins, version 21.0pr7

- HTMLViewer JavaScript communication for Xojo

- New in the MBS Xojo Plugins 20.0

- MBS Xojo Plugins, version 20.0pr5

- Upgrading our HTMLViewer functions for Internet Explorer

Xojo Developer Magazine

- 19.2, page 10: News

- 18.3, page 53: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

19.11.2   Methods

19.11.3   Alert(Message as string)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box containing an application-defined message.
Example:

dim w as IEWindowMBS = HTMLViewer1.IEWindowMBS
w.Alert("Hello Alert!")

19.11.4 Blur

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Causes the element to lose focus and fires the onblur event.

19.11.5 Confirm(Message as string) as Boolean

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Displays a confirmation dialog box that contains an optional message as well as OK and Cancel buttons.
Example:

dim w as IEWindowMBS = HTMLViewer1.IEWindowMBS

dim b as Boolean = w.Confirm("Like Xojo?")
Break

Notes: Message: that specifies the message to display in the confirmation dialog box.
If no value is provided, the dialog box does not contain a message.

The title bar of the confirmation dialog box cannot be changed.

19.11.6 Constructor(DesktopHTMLViewer as DesktopHTMLViewer)

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

Function: Creates new web window object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.
See also:

- 19.11.7 Constructor(HTMLViewer as HTMLViewer) 575

19.11.7 Constructor(HTMLViewer as HTMLViewer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Creates new web window object for this HTMLViewer.
Notes: For browser based on Internet Explorer on Windows.
Raises exception on failure.
See also:

19.11.8   ExecScript(Code as string, language as String)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Executes the specified script in the provided language.
Example:

dim w as IEWindowMBS = HTMLViewer1.IEWindowMBS

w.ExecScript("alert('Hello');", "JavaScript")

Notes: code: specifies the code to be executed.
language: specifies the language in which the code is executed. The language defaults to JScript.

Script executed through the execScript method can access all global variables available to the calling script. This can be useful when you want the functionality of another scripting language that would not otherwise be available in JScript, such as the Microsoft Visual Basic Scripting Edition (VBScript) MsgBox function.

19.11.9   Frames as IEWindowMBS()

Plugin Version: 21.0, Platform: Windows, Targets: Desktop only.

Function: Queries the collection of frames.
Example:

Dim win As IEWindowMBS = HTMLViewer1.IEWindowMBS
Dim frames() As IEWindowMBS = win.Frames

For Each frame As IEWindowMBS In frames
Dim doc As IEDocumentMBS = frame.Document

Dim URL As String = doc.URL
Dim Name As String = doc.NameProp
Dim HTMLText As String = doc.HTMLText

Break // read in debugger
Next

Notes: MBS Plugin will return the window objects for each frame as an array.

19.11.10 MoveBy(x as integer, y as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Moves the screen position of the window by the specified x and y offset values.
Notes: x: long that specifies the horizontal scroll offset in pixels. The value can be either positive or negative.
y: long that specifies the vertical scroll offset in pixels. The value can be either positive or negative.

19.11.11 MoveTo(x as integer, y as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Moves the screen position of the upper-left corner of the window to the specified x and y position.
Notes: x: long that specifies the horizontal scroll offset in pixels. The value can be either positive or negative.
y: long that specifies the vertical scroll offset in pixels. The value can be either positive or negative.

19.11.12 Navigate(URL as string)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Loads the specified URL to the current window.
Notes: URL: that specifies the URL to display.

19.11.13 Print

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Opens the normal print dialog for the Internet Explorer.
Notes: Raises exception on failure.

19.11.14 RunJavaScript(JavaScript as string)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Executes the specified script in the provided language.
Example:

dim w as IEWindowMBS = HTMLViewer1.IEWindowMBS

w.RunJavaScript("alert('Hello');")

Notes: Same as ExecScript with language JavaScript.
Raises exception on failure.

The IE API does not allow to return values from Javascript. So you need to store your result in window.title and access it later using IETitleMBS.

IERunJavaScriptMBS fails if the htmlviewer is empty. You can load a dummy page like above.

You can use JavaScriptEngineMBS class to execute JavaScript without HTMLViewer in our own cross platform JavaScript engine.

See IEDocumentMBS.Evaluate for newer version which can return result.

19.11.15   Scroll(x as integer, y as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Causes the window to scroll to the specified x- and y-offset at the upper-left corner of the window.
Notes: x: long that specifies the horizontal scroll offset, in pixels.
y: long that specifies the vertical scroll offset, in pixels.

This method is provided for backward compatibility only. The recommended way to scroll a window is to use the scrollTo method.

19.11.16   ScrollBy(x as integer, y as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Causes the window to scroll relative to the current scrolled position by the specified x- and y-pixel offset.
Example:

dim w as IEWindowMBS = HTMLViewer1.IEWindowMBS

w.ScrollBy 0, 1000

Notes: x: long that specifies the horizontal scroll offset, in pixels. Positive values scroll the window right, and negative values scroll it left.

y: long that specifies the vertical scroll offset, in pixels. Positive values scroll the window down, and negative values scroll it up.

19.11.17 ScrollTo(x as integer, y as integer)

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Scrolls the window to the specified x- and y-offset.
Notes: x: long that specifies the horizontal scroll offset, in pixels.
y: long that specifies the vertical scroll offset, in pixels.

19.11.18 Properties

19.11.19 DefaultStatus as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or retrieves the default message displayed in the status bar at the bottom of the window.
Notes: (Read and Write property)

19.11.20 Document as IEDocumentMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop only.

Function: Queries the document for this window.
Notes: May be nil if no document is loaded yet.
(Read only property)

19.11.21 Handle as Integer

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

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

19.11.22 Name as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or retrieves a value that indicates the window name.
Notes: (Read and Write property)

19.11.23   Status as String

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Sets or retrieves the message in the status bar at the bottom of the window.
Notes: (Read and Write property)

19.12 Globals

19.12.1 IEClearBrowserSessionMBS as boolean

Plugin Version: 18.5, Platform: Windows, Targets: Desktop only.

Function: Ends browser session.
Notes: For HTMLViewer using Internet Explorer engine. If you close one by closing the window with HTMLViewer control, this method should end session and a new window would start fresh again.

Flushes entries not in use from the password cache on the hard disk drive. Also resets the cache time used when the synchronization mode is once-per-session.
This should clear current session cookies.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.5

- MBS Xojo Plugins, version 18.5pr7

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

Chapter 20

Image Capture

20.1 class WIADataCallbackMBS

20.1.1 class WIADataCallbackMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Provides an application callback mechanism during data transfers from Windows Image Acquisition (WIA) hardware devices to applications.
Notes: Works on Windows 2000 Professional, Windows XP and Windows Server 2003.

20.1.2 Properties

20.1.3 Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.1.4 Events

20.1.5 BandedDataCallback(message as Integer, Status as Integer, Percent-Complete as Integer, Offset as Integer, Length as Integer, Buffer as memoryblock) as Integer

Plugin Version: 10.3, Platform: Windows, Targets: .

Function: Provides data transfer status notifications.

Notes: Windows Image Acquisition (WIA) data transfer methods of the WiaDataTransfer interface periodically call this method.

Message: Specifies a constant that indicates the reason for the callback. Can be one of the kMessage* constants.

Status: Specifies a constant that indicates the status of the WIA device. Can be set to a combination of the kStatus* constants.

PercentComplete: Specifies the percentage of the total data that has been transferred so far.

Offset: Specifies an offset, in bytes, from the beginning of the buffer where the current band of data begins.

Length: Specifies the length, in bytes, of the current band of data.

Buffer: The data buffer.

Lasterror is set.

Your application must provide the BandedDataCallback event. This event is periodically invoked by the data transfer methods of the WiaDataTransferMBS interface. It provides status messages to the application during the data transfer. By returning false, your program can also use this method to prematurely terminate the data transfer.

When this method is invoked, the Message parameter will contain the reason for the call. Not all parameters will contain data on all calls. For example, when BandedDataCallback is invoked with a message of kMessageTermination, it should not attempt to use the values in the Buffer, Offset, and Length parameters.

If the value of Message is kMessageData, the buffer contains a band of image data. The Offset parameter contains an offset in bytes from the beginning of the buffer where the current band of data begins. The Length parameter specified the length in bytes of the current band of data.

During calls where Message is set to kMessageData or kMessageStatus, the Status parameter contains a valid value. Its contents should not be used when Message contains other values.

If Message is kMessageDataHeader, the Buffer parameter points to a WIA_DATA_CALLBACK_HEADER structure.

When an error has occurred during an image data transfer, the driver sets Message to IT_MSG_DEVICE_STATUS. The proxy callback object calls ReportStatus, which handles the error and displays messages to the user.

20.1.6   Constants

Constants

Constant	Value	Description
kMessageData	2	One of the constants for the message parameter in the BandedDataCallback callback.
		The WIA system is transferring data to the application.
kMessageDataHeader	1	One of the constants for the message parameter in the BandedDataCallback callback.
		The application is receiving a header prior to receiving the actual data.
kMessageFilePreviewData	6	One of the constants for the message parameter in the BandedDataCallback callback.
		The WIA system is transferring preview data to the application.
kMessageFilePreviewDataHeader	7	One of the constants for the message parameter in the BandedDataCallback callback.
		The application is receiving a header prior to receiving the actual preview data.
kMessageNewPage	5	One of the constants for the message parameter in the BandedDataCallback callback.
		The data transfer is beginning a new page.
kMessageStatus	3	One of the constants for the message parameter in the BandedDataCallback callback.
		This invocation of the callback is sending only status information.
kMessageTermination	4	One of the constants for the message parameter in the BandedDataCallback callback.
		The data transfer is complete.
kStatusProcessingData	2	One of the constants for the status parameter in the BandedDataCallback callback.
		Data is currently being processed.
kStatusTransferFromDevice	1	One of the constants for the status parameter in the BandedDataCallback callback.
		Data is currently being transferred from the WIA device.
kStatusTransferToClient	4	One of the constants for the status parameter in the BandedDataCallback callback.
		Data is currently being transferred to the client's data buffer.

20.2   class WIADataTransferInfoMBS

20.2.1   class WIADataTransferInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADataTransferInfoMBS class is used by applications to describe the buffer used to retrieve bands of data from Windows Image Acquisition (WIA) devices.
Notes: It is primarily used in conjunction with the methods of the IWiaDataTransfer interface.

20.2.2   Properties

20.2.3   BufferSize as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size in bytes of the buffer that is used for the data transfer.
Notes: (Read and Write property)

20.2.4   DoubleBuffer as Boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Contains true if the device is double buffered, false if the device is not double buffered.
Notes: (Read and Write property)

20.2.5   Section as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies an optional handle to a shared section of memory allocated by the application. If this member is set to nil, GetBandedData allocates the shared memory itself.
Notes: (Read and Write property)

20.2.6   Size as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Contains the size of this structure.
Notes: (Read and Write property)

20.3   class WIADataTransferMBS

20.3.1   class WIADataTransferMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADataTransferMBS interface is a high performance data transfer interface.
Notes: This interface supports a shared memory window to transfer data from the device object to the application, and eliminates unnecessary data copies during marshalling. A callback mechanism is provided in the form of the WiaDataCallbackMBS interface. It enables applications to obtain data transfer status notification, transfer data from the Windows Image Acquisition (WIA) device to the application, and cancel pending data transfers.

For Windows Vista applications, use IWiaTransfer instead of IWiaDataTransfer.

20.3.2   Methods

20.3.3   EnumerateFormatInfo as WIAFormatInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an enumerator for the format information.
Notes: Lasterror is set.

20.3.4   GetBandedData(DataTransInfo as WIADataTransferInfoMBS, DataCallback as WIADataCallbackMBS)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetBandedData method transfers a band of data from a hardware device to an application.
Notes: For efficiency, applications retrieve data from Windows Image Acquisition (WIA) hardware devices in successive bands.
Lasterror is set.

20.3.5   GetDataFile(DataCallback as WIADataCallbackMBS) as folderitem

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetData method retrieves complete files from a Windows Image Acquisition (WIA) device.
Notes: Lasterror is set.
Returns the folderitem for the new file. Copy or load the file as this temporary file is deleted as soon as the

object is destroyed.

20.3.6   GetDataPath(DataCallback as WIADataCallbackMBS) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetData method retrieves complete files from a Windows Image Acquisition (WIA) device.
Notes: Lasterror is set.
Returns the folderitem for the new file. Copy or load the file as this temporary file is deleted as soon as the object is destroyed.

20.3.7   GetExtendedTransferInfo as WIAExtendedTransferInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetExtendedTransferInfo retrieves extended information relating to data transfer buffers in the case of banded data transfers.
Notes: Applications typically use this method to retrieve driver recommended settings for minimum buffer size, maximum buffer size, and optimal buffer size for banded data transfers.

Lasterror is set.

20.3.8   QueryGetData as WIAFormatInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The QueryGetData method is used by applications to query a Windows Image Acquisition (WIA) device to determine what types of data formats it supports.
Notes: Lasterror is set.

20.3.9   Properties

20.3.10   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.3.11 Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.4   class WIADeviceCapabilitiesEnumeratorMBS

20.4.1   class WIADeviceCapabilitiesEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADeviceCapabilitiesEnumeratorMBS class enumerates the currently available Windows Image Acquisition (WIA) hardware device capabilities.
Notes: Device capabilities include commands and events that the device supports.

20.4.2   Methods

20.4.3   Clone as WIADeviceCapabilitiesEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a copy of the WIADeviceCapabilitiesEnumeratorMBS object.
Notes: Lasterror is set.

20.4.4   Count as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the number of items in the enumeration.
Notes: Lasterror is set.

20.4.5   NextItem as WIADeviceCapabilitiesMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next item in the enumeration.
Notes: Lasterror is set.

20.4.6   Reset

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumeration.
Notes: Lasterror is set.

20.4.7   Skip(celt as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips the given number of entries in the enumeration.
Notes: Lasterror is set.

20.4.8   Properties

20.4.9   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.4.10   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.5   class WIADeviceCapabilitiesMBS

20.5.1   class WIADeviceCapabilitiesMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for device capabilities.

20.5.2   Properties

20.5.3   Commandline as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a string that represents command line arguments.
Notes: (Read and Write property)

20.5.4   Description as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a string that contains a description of the capability that is displayed to the user.
Notes: (Read and Write property)

20.5.5   Flags as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Flags for this capability
Notes: (Read and Write property)

20.5.6   GUID as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a GUID that identifies the device capability.
Notes: This member can be set to any of the values specified in WIAItemMBS constants for Device Commands (kCommand*) or WIA Event Identifiers (kEvent*).
(Read and Write property)

20.5.7 Icon as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a string that represents the location and resource ID of the icon that represents this capability or handler.
Notes: The string must be of the following form: drive:\path\module,n, where n is the icon's negated resource ID (that is, if the resource ID of the icon is 100, then n is -100).
(Read and Write property)

20.5.8 Name as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a string that contains a short version of the capability name.
Notes: (Read and Write property)

20.6   class WIADeviceInfoEnumeratorMBS

20.6.1   class WIADeviceInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADeviceInfoEnumeratorMBS class enumerates the currently available Windows Image Acquisition (WIA) hardware devices and their properties.
Notes: Device information properties describe the installation and configuration of WIA hardware devices.

20.6.2   Methods

20.6.3   Clone as WIADeviceInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a copy of the enumerator.
Notes: Lasterror is set.

20.6.4   Count as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the number of items in the enumeration.
Notes: Lasterror is set.

20.6.5   NextItem as WIAPropertyStorageMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next item in the enumeration.
Notes: Lasterror is set.

20.6.6   Reset

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumeration.
Notes: Lasterror is set.

20.6.7 Skip(celt as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips the given number of entries in the enumeration.
Notes: Lasterror is set.

20.6.8 Properties

20.6.9 Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.6.10 Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.7   class WIADeviceManager1MBS

20.7.1   class WIADeviceManager1MBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADeviceManager1MBS interface is used to create and manage image acquisition devices and to register to receive device events.
Notes: WIA 1.x is available on Windows 2000 and newer
Blog Entries

- MBS Real Studio Plugins, version 12.1pr1

- MBS Plugins 10.3 Release Notes

- Check the WIA Plugin part today

- MBS REALbasic Plugins, version 10.3pr4

20.7.2   Methods

20.7.3   Constructor

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new WIA 1.0 manager object.
Example:

dim DeviceManager as new WIADeviceManager1MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
MsgBox "OK"
end if

20.7.4   CreateDevice(DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a hierarchical tree of WiaItem objects for a Windows Image Acquisition device.
Notes: DeviceID: Specifies the unique identifier of the WIA device.

Lasterror is set.

Applications use the CreateDevice method to create a device object for the WIA devices specified by the DeviceID parameter.

Returns the WIAItemMBS object for the root item. Applications can use this tree of objects to control and retrieve data from the WIA device.

20.7.5 EnumDeviceInfo(flags as Integer = &h10) as WIADeviceInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an enumerator of property information for each available Windows Image Acquisition device.
Example:

dim DeviceManager1 as new WIADeviceManager1MBS

// Enumerate all local devices
dim e as WIADeviceInfoEnumeratorMBS = DeviceManager1.EnumDeviceInfo(DeviceManager1.kEnumLocal)
if e<>Nil then

dim p as WIAPropertyStorageMBS = e.NextItem
while p<>Nil

// display the name of the device in a listbox
Listbox1.AddFolder p.Read(p.kDevicePropertyDevNameString)

p = e.NextItem
wend
end if

Notes: Flags: Specifies the types of WIA devices to enumerate. Should be set to kEnumLocal.
Lasterror is set.

20.7.6 GetImageDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer, Intent as integer, file as folderitem, rootitem as WIAItemMBS=nil)

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

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire an

image from a Windows Image Acquisition (WIA) device and write the image to a specified file.
Notes: Lasterror is set.

This method combines the functionality of SelectDeviceDialog to completely encapsulate image acquisition within a single API call.

parentWindowHandle: Handle of the window that owns the Get Image dialog box.
DeviceType: Specifies which type of WIA device to use. Is set to kDeviceTypeDefault, kDeviceTypeScanner, or kDeviceTypeDigitalCamera.
Flags: Specifies dialog box behavior. Can be set to the following constants: kSelectDeviceNoDefault, kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. Use kIntent* constants.
rootitem: Returns the interface of the hierarchical tree of WiaItem objects returned by CreateDevice.
file: Specifies the name of the file to which the image data is written.

Invoking this method displays a dialog box that enables users to acquire images. It can also display the Select Device dialog box created by the SelectDeviceDlg method.

If the application passes nil for the value of the rootitem parameter, GetImageDlg displays the Select Device dialog box that lets the user select the WIA input device. If the application specifies a WIA input device by passing a pointer to the device's item tree through the pItemRoot parameter, GetImageDlg does not display the Select Device dialog box. Instead, it will use the specified input device to acquire the image.

When using the Select Device dialog box, applications can specify types of WIA input devices. To do so, they must set the rootitem parameter to NULL and pass the appropriate constants through the DeviceType parameter. If more than one device of the specified type is present, the GetImageDlg displays the Select Device dialog box to let the user select which device will be used.

If GetImageDlg finds only one matching device, it will not display the Select Device dialog box. Instead, it will select the matching device. You can override this behavior and force GetImageDlg to display the Select Device dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.

The dialog must have sufficient rights to the folder for file that it can save the file with a unique file name. The folder should also be protected with an access control list (ACL) because it contains user data.
See also:

20.7.7 GetImageDialog(parentWindow as window, DeviceType as Integer, Flags as Integer, Intent as Integer, file as folderitem, rootitem as WIAItemMBS=nil)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire an image from a Windows Image Acquisition (WIA) device and write the image to a specified file.
Notes: Lasterror is set.

This method combines the functionality of SelectDeviceDialog to completely encapsulate image acquisition within a single API call.

parentWindowHandle: Handle of the window that owns the Get Image dialog box.
DeviceType: Specifies which type of WIA device to use. Is set to kDeviceTypeDefault, kDeviceTypeScanner, or kDeviceTypeDigitalCamera.
Flags: Specifies dialog box behavior. Can be set to the following constants: kSelectDeviceNoDefault, kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. Use kIntent* constants.
rootitem: Returns the interface of the hierarchical tree of WiaItem objects returned by CreateDevice.
file: Specifies the name of the file to which the image data is written.

Invoking this method displays a dialog box that enables users to acquire images. It can also display the Select Device dialog box created by the SelectDeviceDlg method.

If the application passes nil for the value of the rootitem parameter, GetImageDlg displays the Select Device dialog box that lets the user select the WIA input device. If the application specifies a WIA input device by passing a pointer to the device's item tree through the pItemRoot parameter, GetImageDlg does not display the Select Device dialog box. Instead, it will use the specified input device to acquire the image.

When using the Select Device dialog box, applications can specify types of WIA input devices. To do so, they must set the rootitem parameter to NULL and pass the appropriate constants through the DeviceType parameter. If more than one device of the specified type is present, the GetImageDlg displays the Select Device dialog box to let the user select which device will be used.

If GetImageDlg finds only one matching device, it will not display the Select Device dialog box. Instead, it will select the matching device. You can override this behavior and force GetImageDlg to display the Select Device dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.

The dialog must have sufficient rights to the folder for file that it can save the file with a unique file name. The folder should also be protected with an access control list (ACL) because it contains user data.

See also:

20.7.8   GetImageDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer, Intent as Integer, file as folderitem, rootitem as WIAItemMBS=nil)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire an image from a Windows Image Acquisition (WIA) device and write the image to a specified file.
Notes: Lasterror is set.

This method combines the functionality of SelectDeviceDialog to completely encapsulate image acquisition within a single API call.

parentWindowHandle: Handle of the window that owns the Get Image dialog box.
DeviceType: Specifies which type of WIA device to use. Is set to kDeviceTypeDefault, kDeviceTypeScanner, or kDeviceTypeDigitalCamera.
Flags: Specifies dialog box behavior. Can be set to the following constants: kSelectDeviceNoDefault, kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. Use kIntent* constants.
rootitem: Returns the interface of the hierarchical tree of WiaItem objects returned by CreateDevice.
file: Specifies the name of the file to which the image data is written.

Invoking this method displays a dialog box that enables users to acquire images. It can also display the Select Device dialog box created by the SelectDeviceDlg method.

If the application passes nil for the value of the rootitem parameter, GetImageDlg displays the Select Device dialog box that lets the user select the WIA input device. If the application specifies a WIA input device by passing a pointer to the device's item tree through the pItemRoot parameter, GetImageDlg does not display the Select Device dialog box. Instead, it will use the specified input device to acquire the image.

When using the Select Device dialog box, applications can specify types of WIA input devices. To do so, they must set the rootitem parameter to NULL and pass the appropriate constants through the DeviceType parameter. If more than one device of the specified type is present, the GetImageDlg displays the Select Device dialog box to let the user select which device will be used.

If GetImageDlg finds only one matching device, it will not display the Select Device dialog box. Instead, it

will select the matching device. You can override this behavior and force GetImageDlg to display the Select Device dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.

The dialog must have sufficient rights to the folder for file that it can save the file with a unique file name. The folder should also be protected with an access control list (ACL) because it contains user data.
See also:

20.7.9   SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as WIAItemMBS

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or "" if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.7.10   SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer, byref DeviceID as string) as WIAItemMBS

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.7.9 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as WIAItemMBS 601

- 20.7.11 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer) as WIAItemMBS 603

- 20.7.12 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS 604

- 20.7.13 SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as WIAItemMBS 605

- 20.7.14 SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS 606

20.7.11 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Example:

dim DeviceManager as new WIADeviceManager1MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
dim it as WIAItemMBS = DeviceManager.SelectDeviceDialog(window1, DeviceManager.kDeviceTypeDefault, DeviceManager.kSelectDeviceNoDefault)

if it<>Nil then
dim p as WIAPropertyStorageMBS = it.PropertyStorage
dim name as string = p.Read(p.kItemPropertyItemNameString)
MsgBox name
end if
end if

Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault

DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.7.12   SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.

Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault

DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or "" if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.7.9 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as WIAItemMBS   601

- 20.7.10 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer, byref DeviceID as string) as WIAItemMBS   602

- 20.7.11 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer) as WIAItemMBS   603

- 20.7.13 SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as WIAItemMBS   605

- 20.7.14 SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS   606

20.7.13   SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of

possible values.

Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault

DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.

Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.

See also:

20.7.14   SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.

DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or "" if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.7.9 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as WIAItemMBS   601

- 20.7.10 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer, byref DeviceID as string) as WIAItemMBS   602

- 20.7.11 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer) as WIAItemMBS   603

- 20.7.12 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS   604

- 20.7.13 SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as WIAItemMBS   605

20.7.15   SelectDeviceDialogID(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as string

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.7.16 SelectDeviceDialogID(parentWindow as window, DeviceType as Integer, Flags as Integer) as string   608

- 20.7.17 SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string   609

20.7.16   SelectDeviceDialogID(parentWindow as window, DeviceType as Integer, Flags as Integer) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.7.15 SelectDeviceDialogID(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as string 607

- 20.7.17 SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string 609

20.7.17 SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device

types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.7.18   Properties

20.7.19   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 1.x.
(Read and Write property)

20.7.20   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.7.21   Constants

Constants

Constant	Value	Description
kDeviceDialogSingleImage	2	One of the constants for the GetImageDialog method. Restrict image selection to a single image in the device image acquisition dialog box. Only for WIA 2.x.
kDeviceDialogUseCommonUI	4	One of the constants for the GetImageDialog method. Use the system UI, if available, rather than the vendor-supplied UI. If the system UI is not available, the vendor UI is used. If neither UI is available, the function returns E_NOTIMPL.
kDeviceTypeDefault	0	One of the device type constants.
kDeviceTypeDigitalCamera	2	One of the device type constants.
kDeviceTypeScanner	1	One of the device type constants.
kDeviceTypeStreamingVideo	3	One of the device type constants.
kEnumAll	15	One of the constants for EnumDeviceInfo flags parameter. All devices are enumerated, both locally and remote, including inactive (disconnected) devices and legacy STI-only devices.
kEnumLocal	16	One of the constants for EnumDeviceInfo flags parameter. Only locally connected active scanner devices are enumerated.
kIntentBestPreview	&h40000	One of the intent constants for GetImageDialog. Specifies the best quality preview.
kIntentImageTypeColor	1	One of the intent constants for GetImageDialog. Preset properties for color content.
kIntentImageTypeGrayscale	2	One of the intent constants for GetImageDialog. Preset properties for grayscale content.
kIntentImageTypeMask	&hF	One of the intent constants for GetImageDialog. Mask for all of the image type flags.
kIntentImageTypeText	4	One of the intent constants for GetImageDialog. Preset properties for text content.
kIntentMaximizeQuality	&h20000	One of the intent constants for GetImageDialog. Preset properties to maximize image quality.
kIntentMinimizeSize	&h10000	One of the intent constants for GetImageDialog. Preset properties to minimize image size.
kIntentNone	0	One of the intent constants for GetImageDialog. Default value. Do not preset any properties.
kIntentSizeMask	&hF0000	One of the intent constants for GetImageDialog. Mask for all of the size/quality flags.
kSelectDeviceNoDefault	1	One of the constants for the GetImageDialog method. Force this method to display the Select Device dialog box. Only for WIA 2.x.

20.8   class WIADeviceManager2MBS

20.8.1   class WIADeviceManager2MBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIADeviceManager2MBS interface is used to create and manage image acquisition devices and to register to receive device events.
Notes: WIA 2.x is available on Windows Vista and newer.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr1

- MBS Plugins 10.3 Release Notes

- Check the WIA Plugin part today

- MBS REALbasic Plugins, version 10.3pr4

20.8.2   Methods

20.8.3   Constructor

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new WIA 2.0 manager object.
Example:

dim DeviceManager as new WIADeviceManager2MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
MsgBox "OK"
end if

20.8.4   CreateDevice(DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a hierarchical tree of WiaItem objects for a Windows Image Acquisition device.
Notes: DeviceID: Specifies the unique identifier of the WIA device.

Lasterror is set.

Applications use the CreateDevice method to create a device object for the WIA devices specified by the DeviceID parameter.

Returns the WIAItemMBS object for the root item. Applications can use this tree of objects to control and retrieve data from the WIA device.

20.8.5   EnumDeviceInfo(flags as Integer = &h10) as WIADeviceInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an enumerator of property information for each available Windows Image Acquisition device.
Example:

dim DeviceManager2 as new WIADeviceManager2MBS

// Enumerate all local devices
dim e as WIADeviceInfoEnumeratorMBS = DeviceManager2.EnumDeviceInfo(DeviceManager2.kEnumLocal)
if e<>Nil then

dim p as WIAPropertyStorageMBS = e.NextItem
while p<>Nil

// display the name of the device in a listbox
Listbox1.AddFolder p.Read(p.kDevicePropertyDevNameString)

p = e.NextItem
wend
end if

Notes: Flags: Specifies the types of WIA devices to enumerate. Should be set to kEnumLocal or kEnumAll. Lasterror is set.

20.8.6   GetImageDialog(Flags as integer, DeviceID as string, parentWindow as DesktopWindow, FolderName as String, Filename as String, byref item as WIAItemMBS) as string()

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

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire

an image from a Windows Image Acquisition (WIA) 2.0 device and write the image to a specified file. This method extends the functionality of SelectDeviceDlg to encapsulate image acquisition within a single API call.
Notes: Lasterror is set.

Flags: Specifies dialog box behavior. Can be set to the following values: kDeviceDialogUseCommonUI
DeviceID: Specifies the scanner to use.
parentWindowHandle: A handle of the window that owns the Get Image dialog box.
FolderName: Specifies the name of the folder ito store the scanned files in.
Filename: Specifies the name of the file to write the image data to.
item: The variable to return the WiaItem that the images were scanned from.

Returns an array with paths to the files that have been scanned.

If the application passes an emapty string for the value of the DeviceID parameter, GetImageDialog displays the Select Device dialog box so that the user can select the WIA 2.0 input device.

Use a menu item named From scanner on the File menu so that device and image selections are available in your application.

The dialog box must have sufficient rights to FolderName so that it can save the files with unique file names. Protect the folder with an access control list (ACL) because it contains user data.
See also:

- 20.8.7 GetImageDialog(Flags as Integer, DeviceID as string, parentWindow as window, FolderName as String, Filename as String, byref item as WIAItemMBS) as string()   614

- 20.8.8 GetImageDialog(Flags as Integer, DeviceID as string, parentWindowHandle as Integer, Folder-Name as String, Filename as String, byref item as WIAItemMBS) as string()   615

20.8.7   GetImageDialog(Flags as Integer, DeviceID as string, parentWindow as window, FolderName as String, Filename as String, byref item as WIAItemMBS) as string()

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire an image from a Windows Image Acquisition (WIA) 2.0 device and write the image to a specified file. This method extends the functionality of SelectDeviceDlg to encapsulate image acquisition within a single API call.
Notes: Lasterror is set.

Flags: Specifies dialog box behavior. Can be set to the following values: kDeviceDialogUseCommonUI
DeviceID: Specifies the scanner to use.

parentWindowHandle: A handle of the window that owns the Get Image dialog box.
FolderName: Specifies the name of the folder ito store the scanned files in.
Filename: Specifies the name of the file to write the image data to.
item: The variable to return the WiaItem that the images were scanned from.

Returns an array with paths to the files that have been scanned.

If the application passes an emapty string for the value of the DeviceID parameter, GetImageDialog displays the Select Device dialog box so that the user can select the WIA 2.0 input device.

Use a menu item named From scanner on the File menu so that device and image selections are available in your application.

The dialog box must have sufficient rights to FolderName so that it can save the files with unique file names. Protect the folder with an access control list (ACL) because it contains user data.
See also:

20.8.8   GetImageDialog(Flags as Integer, DeviceID as string, parentWindowHandle as Integer, FolderName as String, Filename as String, byref item as WIAItemMBS) as string()

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetImageDialog method displays one or more dialog boxes that enable a user to acquire an image from a Windows Image Acquisition (WIA) 2.0 device and write the image to a specified file. This method extends the functionality of SelectDeviceDlg to encapsulate image acquisition within a single API call.
Notes: Lasterror is set.

Flags: Specifies dialog box behavior. Can be set to the following values: kDeviceDialogUseCommonUI
DeviceID: Specifies the scanner to use.
parentWindowHandle: A handle of the window that owns the Get Image dialog box.
FolderName: Specifies the name of the folder ito store the scanned files in.
Filename: Specifies the name of the file to write the image data to.
item: The variable to return the WiaItem that the images were scanned from.

Returns an array with paths to the files that have been scanned.

If the application passes an empty string for the value of the DeviceID parameter, GetImageDialog displays the Select Device dialog box so that the user can select the WIA 2.0 input device.

Use a menu item named From scanner on the File menu so that device and image selections are available in your application.

The dialog box must have sufficient rights to FolderName so that it can save the files with unique file names. Protect the folder with an access control list (ACL) because it contains user data.
See also:

20.8.9   SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as WIAItemMBS

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.8.10 SelectDeviceDialog(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer, byref DeviceID as string) as WIAItemMBS

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.

See also:

20.8.11 SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Example:

dim DeviceManager as new WIADeviceManager2MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
dim it as WIAItemMBS = DeviceManager.SelectDeviceDialog(window1, DeviceManager.kDeviceTypeDefault, DeviceManager.kSelectDeviceNoDefault)

if it<>Nil then
dim p as WIAPropertyStorageMBS = it.PropertyStorage
dim name as string = p.Read(p.kItemPropertyItemNameString)
MsgBox name
end if
end if

Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault

DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ”” if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.8.12   SelectDeviceDialog(parentWindow as window, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.

Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or "" if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.8.13   SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of

possible values.

Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault

DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or ""." if it is not needed.

Returns the WIAItem which was selected.

Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.

See also:

20.8.14   SelectDeviceDialog(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer, byref DeviceID as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: parentWindow: Specifies the parent window of the Select Device dialog box.

DeviceType: Specifies which type of WIA 2.0 device to use. See WIA Device Type Specifiers for a list of possible values.
Flags: Specifies the behavior of the dialog box. The value can be one of the following constants: kSelectDeviceNoDefault
DeviceID: Optional, On output, receives a string which contains the device's identifier string. On input, pass the address of a pointer if this information is needed, or "" if it is not needed.

Returns the WIAItem which was selected.
Lasterror is set.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialog method creates a hierarchical tree of IWiaItem2 objects for the device. It returns the WiaItemMBS object of the root item.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialog does not display the Select Device dialog box. Instead it returns the WiaItemMBS tree for the device. You can override this behavior and force SelectDeviceDialog to display the dialog box by specifying kSelectDeviceNoDefault as the value for the Flags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the Select Device dialog box so the user may choose one.

It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.8.15   SelectDeviceDialogID(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as string

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

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.8.16 SelectDeviceDialogID(parentWindow as window, DeviceType as Integer, Flags as Integer) as string   623

- 20.8.17 SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string   624

20.8.16 SelectDeviceDialogID(parentWindow as window, DeviceType as Integer, Flags as Integer) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

- 20.8.15 SelectDeviceDialogID(parentWindow as DesktopWindow, DeviceType as integer, Flags as integer) as string   622

- 20.8.17 SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string   624

20.8.17   SelectDeviceDialogID(parentWindowHandle as Integer, DeviceType as Integer, Flags as Integer) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box that enables the user to select a hardware device for image acquisition.
Notes: Lasterror is set.

parentWindow: Specifies the parent window of the Select Device dialog box.
DeviceType: Specifies which type of WIA device to use. See kDeviceType* constants.
Flags: Specifies the behavior of the dialog box. You can pass the following constant: kSelectDeviceNoDefault

Returns the selected DeviceID.

This method creates and displays the Select Device dialog box so the user can select a WIA device for image acquisition. If a device is successfully selected, the SelectDeviceDialogID method retruns its identifier string to the application.

The application can restrict the devices displayed to the user to particular types by specifying the device

types through the DeviceType parameter. If only one device meets the specification, SelectDeviceDialogID does not display the Select Device dialog box. Instead it passes the device's identifier string to the application without displaying the dialog box. You can override this behavior and force SelectDeviceDialogID to display the dialog box by passing kSelectDeviceNoDefault as the value for the lFlags parameter. If more than one WIA device matches the specification, all matching devices are displayed in the SelectDevice dialog box so the user may choose one.

Note It is recommended that applications make device and image selection available through a menu item named From scanner on the File menu.
See also:

20.8.18 Properties

20.8.19 Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 2.x.
(Read and Write property)

20.8.20 Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.8.21 Constants

Constants

Constant	Value	Description
kDeviceDialogSingleImage	2	One of the constants for the GetImageDialog method.
		Restrict image selection to a single image in the device image acquisition dialog box.
		Only for WIA 2.x.
kDeviceDialogUseCommonUI	4	One of the constants for the GetImageDialog method.
		Use the system UI, if available, rather than the vendor-supplied UI. If the system UI is not available, the vendor UI is used. If neither UI is available, the function returns E_NOTIMPL.
kDeviceTypeDefault	0	One of the device type constants.
kDeviceTypeDigitalCamera	2	One of the device type constants.
kDeviceTypeScanner	1	One of the device type constants.
kDeviceTypeStreamingVideo	3	One of the device type constants.
kEnumAll	15	One of the constants for EnumDeviceInfo flags parameter.
		All devices are enumerated, both locally and remote, including inactive (disconnected) devices and legacy STI-only devices.
kEnumLocal	16	One of the constants for EnumDeviceInfo flags parameter.
		Only locally connected active scanner devices are enumerated.
kIntentBestPreview	&h40000	One of the intent constants for GetImageDialog.
		Specifies the best quality preview.
kIntentImageTypeColor	1	One of the intent constants for GetImageDialog.
		Preset properties for color content.
kIntentImageTypeGrayscale	2	One of the intent constants for GetImageDialog.
		Preset properties for grayscale content.
kIntentImageTypeMask	&hF	One of the intent constants for GetImageDialog.
		Mask for all of the image type flags.
kIntentImageTypeText	4	One of the intent constants for GetImageDialog.
		Preset properties for text content.
kIntentMaximizeQuality	&h20000	One of the intent constants for GetImageDialog.
		Preset properties to maximize image quality.
kIntentMinimizeSize	&h10000	One of the intent constants for GetImageDialog.
		Preset properties to minimize image size.
kIntentNone	0	One of the intent constants for GetImageDialog.
		Default value. Do not preset any properties.
kIntentSizeMask	&hF0000	One of the intent constants for GetImageDialog.
		Mask for all of the size/quality flags.
kSelectDeviceNoDefault	1	One of the constants for the GetImageDialog method.
		Force this method to display the Select Device dialog box.

20.9 class WIAExtendedTransferInfoMBS

20.9.1 class WIAExtendedTransferInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAExtendedTransferInfoMBS class specifies extended transfer information for the GetExtendedTransferInfo method.
Notes: Requires Windows 2000 Professional, Windows XP or Windows Server 2003.

20.9.2 Properties

20.9.3 MaxBufferSize as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Driver-recommended maximum buffer size the application could request in a call to GetBandedData.
Notes: Going over this limit is not detrimental, however, the driver can simply not use the whole buffer and limit each band of data to this maximum size.
(Read and Write property)

20.9.4 MinBufferSize as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Minimum buffer size the application should request in a call to GetBandedData.
Notes: (Read and Write property)

20.9.5 NumBuffers as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: This value is not used and should be ignored.
Notes: (Read and Write property)

20.9.6 OptimalBufferSize as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Driver-recommended buffer size the application should request in a call to GetBandedData.
Notes: (Read and Write property)

20.9.7   Size as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Size of this structure.
Notes: (Read and Write property)

20.10 class WIAFormatInfoEnumeratorMBS

20.10.1 class WIAFormatInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Use the WIAFormatInfoEnumeratorMBS class to enumerate the format and media type information for a device.

20.10.2 Methods

20.10.3 Clone as WIAFormatInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Clones this enumerator.
Notes: Lasterror is set.

20.10.4 Count as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the number of elements stored by this enumerator.
Notes: Lasterror is set.

20.10.5 NextItem as WIAFormatInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next item in the enumeration.

20.10.6 Reset

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumerator.
Notes: Lasterror is set.

20.10.7   Skip(celt as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips the specified number of structures in the enumeration.
Notes: Lasterror is set.

20.10.8   Properties

20.10.9   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.10.10   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.11 class WIAFormatInfoMBS

20.11.1 class WIAFormatInfoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAFormatInfoMBS class specifies valid format and media type pairs for a device.
Notes: Requires Windows 2000 Professional, Windows XP or Windows Server 2003.

20.11.2 Properties

20.11.3 FormatID as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: GUID that identifies the format.
Notes: (Read and Write property)

20.11.4 Tymed as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The media type that corresponds to the guidFormatID member.
Notes: (Read and Write property)

20.12   class WIAGUIDMBS

20.12.1   class WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a Windows unique ID.
Example:

dim w as WIAGUIDMBS = WIAPropertyStorageMBS.kImageFormatTIFF

MsgBox w.DisplayString

Notes: If you need to validate a GUID or UUID, please check the IsGUID function in our FAQ.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.0

- MBS Xojo Plugins, version 18.0pr6

- MBS Plugins 11.1 Release notes

20.12.2   Methods

20.12.3   Constructor

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a GUID with only zeros.
Example:

dim g as new WIAGUIDMBS

MsgBox g.DisplayString

See also:

- 20.12.4 Constructor(value1 as Integer, value2 as Integer, value3 as Integer, value4 as Integer, value5 as Integer, value6 as Integer, value7 as Integer, value8 as Integer, value9 as Integer, value10 as Integer, value11 as Integer, value12 as Integer, value13 as Integer, value14 as Integer, value15 as Integer, value16 as Integer)   633

20.12.4 Constructor(value1 as Integer, value2 as Integer, value3 as Integer, value4 as Integer, value5 as Integer, value6 as Integer, value7 as Integer, value8 as Integer, value9 as Integer, value10 as Integer, value11 as Integer, value12 as Integer, value13 as Integer, value14 as Integer, value15 as Integer, value16 as Integer)

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new GUID with the given byte values.
Example:

dim g as new WIAGUIDMBS(&h14, &h3e, &h4e, &h83, &h64, &h97, &h11, &hd2, &ha2, &h31, &h00, &hc0, &h4f, &ha3, &h18, &h09)

MsgBox g.DisplayString

See also:

20.12.5 DisplayString as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID well formatted.
Example:

dim w as WIAGUIDMBS = WIAPropertyStorageMBS.kImageFormatTIFF

MsgBox w.DisplayString

20.12.6 Equal(other as WIAGUIDMBS) as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether two GUIDs are equal.
Example:

dim w as WIAGUIDMBS = WIAPropertyStorageMBS.kImageFormatTIFF
dim v as WIAGUIDMBS = WIAPropertyStorageMBS.kImageFormatTIFF

if w.Equal(v) then
MsgBox "Equal, right."
else

MsgBox "not equal, a bug."
end if

v = WIAPropertyStorageMBS.kImageFormatBMP

if w.Equal(v) then
MsgBox "Equal, a bug."
else
MsgBox "not equal, right."
end if

Notes: Returns true if both items are equals.

20.12.7   Parse(GUID as String) as WIAGUIDMBS

Plugin Version: 18.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Parses GUID string into a GUID object.

20.12.8   Properties

20.12.9   Byte(index as Integer) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Read or write the byte value.
Example:

dim g as new WIAGUIDMBS

g.Byte(1) = 65

MsgBox str(g.Byte(1)) // shows 65

Notes: (Read and Write computed property)

20.12.10   Data as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The UID as binary string.
Example:

dim w as WIAGUIDMBS = WIAPropertyStorageMBS.kImageFormatTIFF
MsgBox EncodeBase64(w.Data)

Notes: (Read and Write computed property)

20.13   class WIAItemEnumeratorMBS

20.13.1   class WIAItemEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAItemEnumeratorMBS class is used by applications to enumerate WiaItemMBS objects in the tree's current folder.
Example:

Sub EnumerateItems(root as WIAItemMBS)
dim e as WIAItemEnumeratorMBS = Root.EnumerateChildItems

if e<>Nil then
dim it as WIAItemMBS = e.NextItem

while it<>nil

// do something with item

it = e.NextItem
wend

end if
End Sub

Notes: The Windows Image Acquisition (WIA) run-time system represents every WIA hardware device to applications as a hierarchical tree of WiaItemMBS objects.

20.13.2   Methods

20.13.3   Clone as WIAItemEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an additional instance of the WIAItemEnumeratorMBS object.
Notes: Lasterror is set.

20.13.4   Count as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the number of elements stored by this enumerator.

Notes: Lasterror is set.

20.13.5   NextItem as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next item in the enumeration.
Example:

Sub EnumerateItems(root as WIAItemMBS)
dim e as WIAItemEnumeratorMBS = Root.EnumerateChildItems

if e<>Nil then
dim it as WIAItemMBS = e.NextItem

while it<>nil

// do something with item

it = e.NextItem
wend

end if
End Sub

Notes: Lasterror is set.

20.13.6   Reset

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumeration.
Notes: Lasterror is set.

20.13.7   Skip(celt as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips the specified number of items during an enumeration of available WiaItemMBS objects.
Notes: Lasterror is set.

20.13.8   Properties

20.13.9   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 1.x or 2.x.
(Read and Write property)

20.13.10   Handle1 as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 1.x.
(Read and Write property)

20.13.11   Handle2 as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 2.x.
(Read and Write property)

20.13.12   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.14   class WIAItemMBS

20.14.1   class WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for an item.
Notes: Each Windows Image Acquisition (WIA) hardware device is represented to an application as a hierarchical tree of WiaItem objects. The WiaItem interface provides applications with the ability to query devices to discover their capabilities. It also provides access to data transfer interfaces and item properties. In addition, the WiaItem interface provides methods to enable applications to control the device.

This class encapsulates transparently the system classes for WIA 1.x and 2.x.
Blog Entries

- MBS Plugins 10.3 Release Notes

- MBS REALbasic Plugins, version 10.3pr4

20.14.2   Methods

20.14.3   AnalyzeItem

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The AnalyzeItem method causes the Windows Image Acquisition (WIA) hardware device to acquire and try to detect what data types are present.
Notes: This method is used with scanners to detect what type of data is on a page. When an application calls this method, the WIA hardware device driver scans and analyzes the current page. For each data type it detects, it creates an WiaItem object to represent the region on the page the data occupies.

Image processing and OCR software can use this capability to detect graphics and text on a page. This method adds the regions it creates into the WIA device's WiaItem tree. The application can select the individual regions and use the standard data transfer methods to acquire data from them.

If necessary, applications can override the regions created by this method.

Works only on WIA 1.x.
Lasterror is set.

20.14.4   CreateChildItem(ItemFlags as Integer, CreationFlags as Integer, Item-Name as string, FullItemName as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateChildItem method is used by applications to add WiaItem objects to the WiaItem tree of a device.
Notes: ItemFlags: Specifies the WIA item type.
CreationFlags: Specifies how to create the new item. Only for WIA 2.x. Can be 0 to set the default values for the properties of the child. Can be &H40000000 to copy the values of all Read/Write properties from the parent.
ItemName: Specifies the WIA item name, such as "Top". You can think of this parameter as being equivalent to a file name.
FullItemName: Specifies the full WIA item name. You can think of this parameter as equivalent to a full path to a file, such as "003\Root\Top". Only for WIA 1.x.

Lasterror is set.
Returns nil on any error and the new item object on success.

Some WIA hardware devices allow applications to create new items in the WiaItem tree that represents the device. Applications must test the devices to see if they support this capability. Use the EnumerateDeviceCapabilities function to enumerate the current device's capabilities.

If the device allows the creation of new items in the WiaItem tree, invoking CreateChildItem creates a new WiaItem that is a child of the current node.

20.14.5   DataTransfer as WIADataTransferMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a Data transfer object.
Notes: Lasterror is set.
Only for WIA 1.x.
Returns nil on any error.

20.14.6   DeleteItem

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes the current IWiaItem object from the object tree of the device.
Notes: Lasterror is set.
Available on both WIA 1.x and 2.x.

The Windows Image Acquisition (WIA) run-time system represents each WIA hardware device connected to the user's computer as a hierarchical tree of IWiaItem objects. A given WIA device may or may not allow applications to delete IWiaItem objects from its tree. Use the EnumerateDeviceCapabilities function to query the device for item deletion capability.

If the device supports item deletion in its WiaItem tree, invoke the DeleteItem method to remove the WiaItem object. Note that this method will only delete an object after all references to the object have been released.

20.14.7 DeviceCommand(command as WIAGUIDMBS) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Issues a command to a Windows Image Acquisition (WIA) hardware device.
Example:

dim targetItem as WIAItemMBS // your item
dim resultItem as WIAItemMBS
resultItem = targetItem.DeviceCommand(targetItem.kCommandTakePicture)

Notes: Command: Specifies the command to send to the WIA 2.0 device. See kCommand* constants. Works with WIA 1.x and 2.x.

Applications use this method to send WIA commands to hardware devices.

When the application sends the kCommandTakePicture command to the device, the WIA run-time system creates the WiaItem object to represent the image. The DeviceCommand method returns this new WIAItemMBS object.

20.14.8 DeviceDialog(Flags as integer, Win as DesktopWindow, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)

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

Function: Displays a dialog box to the user to prepare for image acquisition.
Notes: Lasterror is set.
Only for WIA 2.x.

Flags: Specifies a set of flags that control the dialog box's operation. The value can be either 0 to represent the default behavior or any of the following flags: kDeviceDialogSingleImage, kDeviceDialogUseCommonUI and kSelectDeviceNoDefault
Win: A handle to the parent window.

FolderName: Specifies the folder name where the files are to be transferred.
Filename: Specifies the template file name.
paths: An array to be filled with the file paths.
items: An array to be filled with the wia item objects.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. It is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaTransferMBS class to acquire the image.
See also:

20.14.9   DeviceDialog(Flags as Integer, Win as window, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box to the user to prepare for image acquisition.
Notes: Lasterror is set.
Only for WIA 2.x.

Flags: Specifies a set of flags that control the dialog box's operation. The value can be either 0 to represent the default behavior or any of the following flags: kDeviceDialogSingleImage, kDeviceDialogUseCommonUI and kSelectDeviceNoDefault
Win: A handle to the parent window.
FolderName: Specifies the folder name where the files are to be transferred.
Filename: Specifies the template file name.
paths: An array to be filled with the file paths.
items: An array to be filled with the wia item objects.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. It is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaTransferMBS class to acquire the image.
See also:

20.14.10   DeviceDialog(Flags as Integer, WindowHandle as Integer, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays a dialog box to the user to prepare for image acquisition.
Notes: Lasterror is set.
Only for WIA 2.x.

Flags: Specifies a set of flags that control the dialog box's operation. The value can be either 0 to represent the default behavior or any of the following flags: kDeviceDialogSingleImage, kDeviceDialogUseCommonUI and kSelectDeviceNoDefault
WindowHandle: A handle to the parent window.
FolderName: Specifies the folder name where the files are to be transferred.
Filename: Specifies the template file name.
paths: An array to be filled with the file paths.
items: An array to be filled with the wia item objects.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. It is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaTransferMBS class to acquire the image.
See also:

20.14.11   DeviceDialog(Win as DesktopWindow, Flags as integer, Intent as integer) as WIAItemMBS()

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

Function: The DeviceDialog method is used by applications to display a dialog box to the user to prepare for image acquisition.
Notes: win: Handle of the parent window of the dialog box.
Flags: Specifies a set of flags that control the dialog box's operation. Can be set to any of the following values: kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. For a list of image intent values, kIntent* constants.

Lasterror is set.
Only for WIA 1.x.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. For instance, this dialog box enables the user to select images to download from a camera. When using a scanner, it is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaDataTransferMBS interface to acquire the image.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.
See also:

20.14.12 DeviceDialog(Win as window, Flags as Integer, Intent as Integer) as WIAItemMBS()

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The DeviceDialog method is used by applications to display a dialog box to the user to prepare for image acquisition.
Notes: win: Handle of the parent window of the dialog box.
Flags: Specifies a set of flags that control the dialog box's operation. Can be set to any of the following values: kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. For a list of image intent values, kIntent* constants.

Lasterror is set.
Only for WIA 1.x.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. For instance, this dialog box enables the user to select images to download from a camera. When using a scanner, it is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaDataTransferMBS interface to acquire the image.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.
See also:

- 20.14.8 DeviceDialog(Flags as integer, Win as DesktopWindow, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS) 641

- 20.14.9 DeviceDialog(Flags as Integer, Win as window, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS) 642

- 20.14.10 DeviceDialog(Flags as Integer, WindowHandle as Integer, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS) 643

- 20.14.11 DeviceDialog(Win as DesktopWindow, Flags as integer, Intent as integer) as WIAItemMBS() 644

- 20.14.13 DeviceDialog(WindowHandle as Integer, Flags as Integer, Intent as Integer) as WIAItemMBS() 645

20.14.13 DeviceDialog(WindowHandle as Integer, Flags as Integer, Intent as Integer) as WIAItemMBS()

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DeviceDialog method is used by applications to display a dialog box to the user to prepare for image acquisition.
Notes: WindowHandle: Handle of the parent window of the dialog box.
Flags: Specifies a set of flags that control the dialog box's operation. Can be set to any of the following values: kDeviceDialogUseCommonUI and kDeviceDialogSingleImage.
Intent: Specifies what type of data the image is intended to represent. For a list of image intent values, kIntent* constants.

Lasterror is set.
Only for WIA 1.x.

This method displays a dialog box to the user that an application uses to gather all the information required for image acquisition. For instance, this dialog box enables the user to select images to download from a camera. When using a scanner, it is also used to specify image scan properties such as brightness and contrast.

After this method returns, the application can use the WiaDataTransferMBS interface to acquire the image.

It is recommended that applications make device and image selection available through a menu item named From scanner or camera on the File menu.
See also:

- 20.14.8 DeviceDialog(Flags as integer, Win as DesktopWindow, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)   641

- 20.14.9 DeviceDialog(Flags as Integer, Win as window, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)   642

- 20.14.10 DeviceDialog(Flags as Integer, WindowHandle as Integer, FolderName as string, Filename as string, paths() as string, items() as WIAItemMBS)   643

- 20.14.11 DeviceDialog(Win as DesktopWindow, Flags as integer, Intent as integer) as WIAItemMBS()   644

- 20.14.12 DeviceDialog(Win as window, Flags as Integer, Intent as Integer) as WIAItemMBS()   645

20.14.14 EnumerateChildItems(CategoryGUID as WIAGUIDMBS=nil) as WIAItemEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates and returns an enumerator object for non-empty folders in a WiaItem tree of a Windows Image Acquisition (WIA) device.
Example:

Sub EnumerateItems(root as WIAItemMBS)
dim e as WIAItemEnumeratorMBS = Root.EnumerateChildItems

```
if e<>Nil then
dim it as WIAItemMBS = e.NextItem

while it<>nil

// do something with item

it = e.NextItem
wend

end if
End Sub
```

Notes: Lasterror is set.
Works for WIA 1.x and WIA 2.x.

CategoryGUID: Specifies a category for which child nodes are enumerated. If nil, then all child nodes are enumerated. This parameter is only used on WIA 2.x.

The WIA run-time system represents each WIA hardware device as a hierarchical tree of WiaItem objects. The EnumerateChildItems method enables applications to enumerate child items in the current item. However, it can only be applied to items that are folders.

If the folder is not empty, it contains a subtree of WiaItem objects. The EnumerateChildItems method enumerates all of the items contained in the folder.

20.14.15 EnumerateDeviceCapabilities(Flags as Integer) as WIADeviceCapabilitiesEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an enumerator that is used to ascertain the commands and events a Windows Image Acquisition (WIA) device supports.
Notes: Flags: Specifies a flag that selects the type of capabilities to enumerate. Can be a combination of kDeviceCommands and kDeviceEvents.

Works for both WIA 1.x and 2.x
Lasterror is set.

Use this method to create an enumerator object to obtain the set of commands and events that a WIA device

supports. You can use the Flags parameter to specify which kinds of device capabilities to enumerate.

20.14.16   FindItemByName(name as string) as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Searches an item's tree of subitems using the name as the search key.
Notes: name: Specifies the name fo the item to search for.

Lasterror is set.
Works with WIA 1.x and 2.x.

This method searches the current item's tree of sub-items using the name as the search key. If FindItemBy-Name finds the item specified by name, it retruns the WiaItem object.

20.14.17   ItemCategory as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets an item's category information.
Notes: Lasterror is set. Only for WIA 2.x.

Every WiaItemMBS object in the hierarchical tree of objects associated with a Windows Image Acquisition (WIA) 2.0 hardware device has a specific category. This method enables applications to identify the category of any item in a hierarchical tree of item objects in a device.

Requires Windows Vista or Windows Server 2008.

20.14.18   ItemType as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets an item's type information.
Example:

```
dim it as WIAItemMBS // your item
if BitwiseAnd(it.ItemType,it.kTypeFolder)=it.kTypeFolder or BitwiseAnd(it.ItemType, it.kTypeHasAttach-
ments)=it.kTypeHasAttachments then
msgbox "may have children."
else
msgbox "no children."
```

end if

Notes: Works with WIA 1.x and 2.x.
Lasterror is set.

Every WiaItemMBS object in the hierarchical tree of objects associated with a Windows Image Acquisition (WIA) 2.0 hardware device has a specific data type. Item objects represent folders and files. Folders contain file objects. File objects contain data acquired by the device such as images and sounds. This method enables applications to identify the type of any item in a hierarchical tree of item objects in a device.

An item may have more than one type. For example, an item that represents an audio file will have the type attributes WiaItemTypeAudio bitwiseor WiaItemTypeFile.

20.14.19 kCategoryFeeder as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFeeder.DisplayString

20.14.20 kCategoryFeederBack as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFeederBack.DisplayString

20.14.21 kCategoryFeederFront as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFeederFront.DisplayString

20.14.22   kCategoryFilm as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFilm.DisplayString

20.14.23   kCategoryFinishedFile as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFinishedFile.DisplayString

20.14.24   kCategoryFlatbed as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFlatbed.DisplayString

20.14.25   kCategoryFolder as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryFolder.DisplayString

20.14.26 kCategoryRoot as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA IPA item category constants
Example:

MsgBox WIAItemMBS.kCategoryRoot.DisplayString

20.14.27 kCommandChangeDocument as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.
Example:

MsgBox WIAItemMBS.kCommandChangeDocument.DisplayString

20.14.28 kCommandDeleteAllItems as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.
Example:

MsgBox WIAItemMBS.kCommandDeleteAllItems.DisplayString

20.14.29 kCommandDiagnostic as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.
Example:

MsgBox WIAItemMBS.kCommandDiagnostic.DisplayString

20.14.30 kCommandSynchronize as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.
Example:

MsgBox WIAItemMBS.kCommandSynchronize.DisplayString

20.14.31   kCommandTakePicture as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.

20.14.32   kCommandUnloadDocument as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA command constants.
Example:

MsgBox WIAItemMBS.kCommandUnloadDocument.DisplayString

20.14.33   ParentItem as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the parent item in the tree that represents a Windows Image Acquisition (WIA) 2.0 hardware device.
Notes: Works only in WIA 2.x. Lasterror is set.
Given any WiaItem object in the object tree of a WIA 2.0 hardware device, the application retrieves a pointer to the parent item by calling this function.

20.14.34   PropertyStorage as WIAPropertyStorageMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries properties for this item.
Example:

Sub ListProperties(it as WIAItemMBS, plist as listbox)
// enumerate properties into a given listbox

```
// clear list
PList.DeleteAllRows

// get properties
dim p as WIAPropertyStorageMBS = it.PropertyStorage
if p<>Nil then
dim e as WIAPropertyEnumeratorMBS = p.Enumerate

if e<>nil then
dim ps as WIAPropertyMBS = e.NextItem

while ps<>Nil
// read the property value
dim v as Variant = p.Read(ps)

// get some identifier string for the listbox, name or id
dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if

PList.AddRow k

if v.Type = v.TypeObject then
if v isa WIAGUIDMBS then
dim g as WIAGUIDMBS = v
PList.Cell(PList.LastIndex,1)=g.DisplayString
else
PList.Cell(PList.LastIndex,1)="? some object" // should never happen
end if
else
PList.Cell(PList.LastIndex,1)=v.StringValue
end if

ps = e.NextItem
wend

end if
end if
End Sub
```

Notes: Lasterror is set.
Returns nil on any error.

20.14.35   RootItem as WIAItemMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the root item of a tree of item objects used to represent a Windows Image Acquisition (WIA) hardware device.
Notes: Lasterror is set.
Works with WIA 1.x and 2.x.

Given any WiaItem object in the object tree of a WIA hardware device, the application retrieves a pointer to the root item by calling this function.

20.14.36   Transfer as WIATransferMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a transfer object for this item.
Notes: Only available on WIA 2.x.
Lasterror is set.
Returns nil on any error.

20.14.37   Properties

20.14.38   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 1.x or 2.x.
(Read and Write property)

20.14.39   Handle1 as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 1.x.
(Read and Write property)

20.14.40 Handle2 as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: The handle for WIA 2.x.
(Read and Write property)

20.14.41 Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.14.42 Constants

Constants

Constant	Value	Description
kDeviceCommands	1	One of the flag constants for the EnumerateDeviceCapabilities function. Enumerate device commands.
kDeviceDialogSingleImage	2	One of the DeviceDialog and ImageDialog flag constants. Only allow one image to be selected
kDeviceDialogUseCommonUI	4	One of the DeviceDialog and ImageDialog flag constants. Give preference to the system-provided UI, if available.
kDeviceEvents	2	One of the flag constants for the EnumerateDeviceCapabilities function. Enumerate device events.
kIntentBestPreview	&h40000	One of the intent constants for GetImageDialog. Specifies the best quality preview.
kIntentImageTypeColor	1	One of the intent constants for GetImageDialog. Preset properties for color content.
kIntentImageTypeGrayscale	2	One of the intent constants for GetImageDialog. Preset properties for grayscale content.
kIntentImageTypeMask	&hF	One of the intent constants for GetImageDialog. Mask for all of the image type flags.
kIntentImageTypeText	4	One of the intent constants for GetImageDialog. Preset properties for text content.
kIntentMaximizeQuality	&h20000	One of the intent constants for GetImageDialog. Preset properties to maximize image quality.
kIntentMinimizeSize	&h10000	One of the intent constants for GetImageDialog. Preset properties to minimize image size.
kIntentNone	0	One of the intent constants for GetImageDialog. Default value. Do not preset any properties.
kIntentSizeMask	&hF0000	One of the intent constants for GetImageDialog. Mask for all of the size/quality flags.
kSelectDeviceNoDefault	1	One of the Select Device Dialog and Image Dialog flag constants.
kTypeAnalyze	&h00000010	One of the WIA item type constants.
kTypeAudio	&h00000020	One of the WIA item type constants.
kTypeBurst	&h00000800	One of the WIA item type constants.
kTypeDeleted	&h00000080	One of the WIA item type constants.
kTypeDevice	&h00000040	One of the WIA item type constants.
kTypeDisconnected	&h00000100	One of the WIA item type constants.
kTypeFile	&h00000002	One of the WIA item type constants.
kTypeFolder	&h00000004	One of the WIA item type constants.
kTypeFree	&h00000000	One of the WIA item type constants.
kTypeGenerated	&h00004000	One of the WIA item type constants.
kTypeHasAttachments	&h00008000	One of the WIA item type constants.
kTypeHPanorama	&h00000200	One of the WIA item type constants.
kTypeImage	&h00000001	One of the WIA item type constants.
kTypeRoot	&h00000008	One of the WIA item type constants.
kTypeStorage	&h00001000	One of the WIA item type constants.
kTypeTransfer	&h00002000	One of the WIA item type constants.
kTypeVideo	&h00010000	One of the WIA item type constants.
kTypeVPanorama	&h00000400	One of the WIA item type constants.

20.15   class WIAPropertyEnumeratorMBS

20.15.1   class WIAPropertyEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a property enumerator.
Example:

```
Sub ListProperties(it as WIAItemMBS, plist as listbox)
// enumerate properties into a given listbox

// clear list
PList.DeleteAllRows

// get properties
dim p as WIAPropertyStorageMBS = it.PropertyStorage
if p<>Nil then
dim e as WIAPropertyEnumeratorMBS = p.Enumerate

if e<>nil then
dim ps as WIAPropertyMBS = e.NextItem

while ps<>Nil
// read the property value
dim v as Variant = p.Read(ps)

// get some identifier string for the listbox, name or id
dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if

PList.AddRow k

if v.Type = v.TypeObject then
if v isa WIAGUIDMBS then
dim g as WIAGUIDMBS = v
PList.Cell(PList.LastIndex,1)=g.DisplayString
else
PList.Cell(PList.LastIndex,1)="? some object" // should never happen
end if
else
PList.Cell(PList.LastIndex,1)=v.StringValue
end if

ps = e.NextItem
wend
```

end if
end if
End Sub

20.15.2   Methods

20.15.3   Clone as WIAPropertyEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a copy of the enumerator.
Notes: Lasterror is set.

20.15.4   NextItem as WIAPropertyMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next item.
Example:

```
Sub ListProperties(it as WIAItemMBS, plist as listbox)
// enumerate properties into a given listbox

// clear list
PList.DeleteAllRows

// get properties
dim p as WIAPropertyStorageMBS = it.PropertyStorage
if p<>Nil then
dim e as WIAPropertyEnumeratorMBS = p.Enumerate

if e<>nil then
dim ps as WIAPropertyMBS = e.NextItem

while ps<>Nil
// read the property value
dim v as Variant = p.Read(ps)

// get some identifier string for the listbox, name or id
dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if
```

PList.AddRow k

if v.Type = v.TypeObject then
if v isa WIAGUIDMBS then
dim g as WIAGUIDMBS = v
PList.Cell(PList.LastIndex,1)=g.DisplayString
else
PList.Cell(PList.LastIndex,1)="? some object" // should never happen
end if
else
PList.Cell(PList.LastIndex,1)=v.StringValue
end if

ps = e.NextItem
wend

end if
end if
End Sub

Notes: Lasterror is set.

20.15.5   Reset

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resets the enumerator.
Notes: Lasterror is set.

20.15.6   Skip(celt as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Skips the next items.
Notes: Lasterror is set.

20.15.7 Properties

20.15.8 Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.15.9 Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.16 class WIAPropertyMBS

20.16.1 class WIAPropertyMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAPropertyMBS class contains data about a single property in a property set. This data is the property ID and type tag, and the optional string name that may be associated with the property.
Example:

```
Sub ListProperties(it as WIAItemMBS, plist as listbox)
// enumerate properties into a given listbox

// clear list
PList.DeleteAllRows

// get properties
dim p as WIAPropertyStorageMBS = it.PropertyStorage
if p<>Nil then
dim e as WIAPropertyEnumeratorMBS = p.Enumerate

if e<>nil then
dim ps as WIAPropertyMBS = e.NextItem

while ps<>Nil
// read the property value
dim v as Variant = p.Read(ps)

// get some identifier string for the listbox, name or id
dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if

PList.AddRow k

if v.Type = v.TypeObject then
if v isa WIAGUIDMBS then
dim g as WIAGUIDMBS = v
PList.Cell(PList.LastIndex,1)=g.DisplayString
else
PList.Cell(PList.LastIndex,1)="? some object" // should never happen
end if
else
PList.Cell(PList.LastIndex,1)=v.StringValue
end if

ps = e.NextItem
```

wend

end if
end if
End Sub

20.16.2   Properties

20.16.3   ID as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A 32-bit identifier that uniquely identifies the property within the property set.
Example:

dim ps as WIAPropertyMBS // your property

dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if

MsgBox k

Notes: All properties within property sets must have unique property identifiers.
(Read and Write property)

20.16.4   Name as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that contains the optional string name associated with the property
Example:

dim ps as WIAPropertyMBS // your property

dim k as string = ps.Name
if len(k)=0 then
k = str(ps.ID)
end if

MsgBox k

Notes: (Read and Write property)

20.16.5   Type as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The property type.
Notes: (Read and Write property)

20.17   class WIAPropertyStorageMBS

20.17.1   class WIAPropertyStorageMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAPropertyStorageMBS class manages the persistent properties of a single property set.
Example:

dim DeviceManager as new WIADeviceManager1MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
dim it as WIAItemMBS = DeviceManager.SelectDeviceDialog(window1, DeviceManager.kDeviceTypeDefault, DeviceManager.kSelectDeviceNoDefault)

if it<>Nil then
dim p as WIAPropertyStorageMBS = it.PropertyStorage
dim name as string = p.Read(p.kItemPropertyItemNameString)
MsgBox name
end if
end if

Notes: Persistent properties consist of information that can be stored persistently in a property set, such as the summary information associated with a file. This contrasts with run-time properties associated with Controls and Automation, which can be used to affect system behavior. Use the methods of the WIAPropertyStorageMBS interface to create or open a persistent property set. An instance of the WIAPropertyStorageMBS interface can manage zero or more WIAPropertyStorageMBS instances.

Each property within a property set is identified by a property identifier (ID), a integer value unique to that set. You can also assign a string name to a property through the WIAPropertyStorageMBS interface.

The automatic conversion to variant supports:
nil, integer, uint32, int64, uint64, single, double, boolean, string and WIAGUIDMBS.

The automatic conversion from variant supports:
integer, boolean, single, double, Int64, string and WIAGUIDMBS.

20.17.2 Methods

20.17.3 Commit(flags as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Saves changes made to a property storage object to the parent storage object.
Notes: See kCommit* flags for the flags parameter.
Lasterror is set.

20.17.4 Count as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the number of properties stored in the property storage.
Notes: Lasterror is set.

20.17.5 Delete(id as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes an item by ID.
Notes: Lasterror is set.
See also:

20.17.6 Delete(name as string)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes an item by name.
Notes: Lasterror is set.
See also:

20.17.7 DeletePropertyName(id as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes specified string names from the current property set.
Notes: Lasterror is set.

id: Property identifier for which string name are to be deleted.

For each property identifier in rgpropid, DeletePropertyName removes any corresponding name-to-property ID mapping. An attempt is silently ignored to delete the name of a property that either does not exist or does not currently have a string name associated with it. This method has no effect on the properties themselves.

20.17.8   Enumerate as WIAPropertyEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enumerates the properties on this property storage.
Notes: Returns nil on any error. Lasterror is set.

20.17.9   kAudioFormatAIFF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kAudioFormatAIFF.DisplayString

20.17.10   kAudioFormatMP3 as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kAudioFormatMP3.DisplayString

20.17.11   kAudioFormatWAV as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kAudioFormatWAV.DisplayString

20.17.12   kAudioFormatWMA as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kAudioFormatWMA.DisplayString

20.17.13   kImageFormatASF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA misc format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatASF.DisplayString

20.17.14   kImageFormatAVI as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatAVI.DisplayString

20.17.15   kImageFormatBMP as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatBMP.DisplayString

20.17.16 kImageFormatCIFF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatCIFF.DisplayString

20.17.17 kImageFormatDPOF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA misc format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatDPOF.DisplayString

20.17.18 kImageFormatEMF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatEMF.DisplayString

20.17.19 kImageFormatExec as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA misc format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatExec.DisplayString

20.17.20 kImageFormatEXIF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatEXIF.DisplayString

20.17.21 kImageFormatFlashPix as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatFlashPix.DisplayString

20.17.22 kImageFormatGIF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatGIF.DisplayString

20.17.23 kImageFormatHTML as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA document format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatHTML.DisplayString

20.17.24 kImageFormatICO as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatICO.DisplayString

20.17.25   kImageFormatJPEG as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatJPEG.DisplayString

20.17.26   kImageFormatJPEG2K as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatJPEG2K.DisplayString

20.17.27   kImageFormatJPEG2KX as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatJPEG2KX.DisplayString

20.17.28   kImageFormatMemoryBMP as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatMemoryBMP.DisplayString

20.17.29 kImageFormatMPG as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA audio format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatMPG.DisplayString

20.17.30 kImageFormatPhotoCD as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatPhotoCD.DisplayString

20.17.31 kImageFormatPICT as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatPICT.DisplayString

20.17.32 kImageFormatPNG as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatPNG.DisplayString

20.17.33 kImageFormatRawRGB as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatRawRGB.DisplayString

20.17.34 kImageFormatRTF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA document format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatRTF.DisplayString

20.17.35 kImageFormatScript as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA misc format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatScript.DisplayString

20.17.36 kImageFormatTIFF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatTIFF.DisplayString

20.17.37 kImageFormatTXT as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA document format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatTXT.DisplayString

20.17.38   kImageFormatUndefined as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatUndefined.DisplayString

20.17.39   kImageFormatUnicode16 as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA misc format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatUnicode16.DisplayString

20.17.40   kImageFormatWMF as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA image format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatWMF.DisplayString

20.17.41   kImageFormatXML as WIAGUIDMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: One of the WIA document format constants.
Example:

MsgBox WIAPropertyStorageMBS.kImageFormatXML.DisplayString

20.17.42 Read(id as Integer) as Variant

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads a property by id.
Notes: Lasterror is set.
See also:

20.17.43 Read(name as string) as Variant

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads a property by name.
Example:

dim DeviceManager as new WIADeviceManager1MBS

if 0 = DeviceManager.Handle then
MsgBox "Failed to initialize device manager."
else
dim it as WIAItemMBS = DeviceManager.SelectDeviceDialog(window1, DeviceManager.kDeviceTypeDefault, DeviceManager.kSelectDeviceNoDefault)

if it<>Nil then
dim p as WIAPropertyStorageMBS = it.PropertyStorage
dim name as string = p.Read(p.kItemPropertyItemNameString)
MsgBox name
end if
end if

Notes: Lasterror is set.
See also:

20.17.44 Read(p as WIAPropertyMBS) as Variant

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads a property by a property specification.
Notes: Lasterror is set.
See also:

20.17.45   ReadPropertyName(id as Integer) as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves any existing string name for the specified property ID.
Notes: Lasterror is set.

20.17.46   Revert

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Revert method discards all changes to the named property set since it was last opened or discards changes that were last committed to the property set.
Notes: This method has no effect on a direct-mode property set.
Lasterror is set.

20.17.47   Write(id as Integer, value as Variant)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes a property by id.
Notes: Lasterror is set.
See also:

20.17.48   Write(name as string, value as Variant, id as Integer = 0)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes a property by name with optional id.
Notes: Lasterror is set.
See also:

20.17.49   Write(p as WIAPropertyMBS, value as Variant)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes a property by property specification.
Notes: Lasterror is set.
See also:

20.17.50   WritePropertyName(id as Integer, name as string)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Assigns string names to a specified array of property IDs in the current property set.
Notes: id: the property ID for which name is to be set.
name: The new name to be assigned to the corresponding property ID in the id parameter. This name may
not exceed 255 characters.

Lasterror is set.

20.17.51   Properties

20.17.52   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.17.53   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.17.54   Constants

Constants

Constant	Value	Description
kCameraDevicePropertyArtist	2090	One of the WI
kCameraDevicePropertyArtistString	"Artist"	One of the WI
kCameraDevicePropertyBatteryStatus	2065	One of the WI
kCameraDevicePropertyBatteryStatusString	"Battery Status"	One of the WI
kCameraDevicePropertyBurstInterval	2075	One of the WI
kCameraDevicePropertyBurstIntervalString	"Burst Interva"	One of the WI
kCameraDevicePropertyBurstNumber	2076	One of the WI
kCameraDevicePropertyBurstNumberString	"Burst Number"	One of the WI
kCameraDevicePropertyCaptureDelay	2082	One of the WI
kCameraDevicePropertyCaptureDelayString	"Capture Delay"	One of the WI
kCameraDevicePropertyCaptureMode	2081	One of the WI
kCameraDevicePropertyCaptureModeString	"Capture Mode"	One of the WI
kCameraDevicePropertyCompressionSetting	2071	One of the WI
kCameraDevicePropertyCompressionSettingString	"Compression Setting"	One of the WI
kCameraDevicePropertyContrast	2080	One of the WI
kCameraDevicePropertyContrastString	"Contrast"	One of the WI
kCameraDevicePropertyCopyrightInfo	2091	One of the WI
kCameraDevicePropertyCopyrightInfoString	"Copyright Info"	One of the WI
kCameraDevicePropertyDigitalZoom	2078	One of the WI
kCameraDevicePropertyDigitalZoomString	"Digital Zoom"	One of the WI
kCameraDevicePropertyDimension	2070	One of the WI
kCameraDevicePropertyDimensionString	"Dimension"	One of the WI
kCameraDevicePropertyEffectMode	2077	One of the WI
kCameraDevicePropertyEffectModeString	"Effect Mode"	One of the WI
kCameraDevicePropertyExposureComp	2053	One of the WI
kCameraDevicePropertyExposureCompString	"Exposure Compensation"	One of the WI
kCameraDevicePropertyExposureIndex	2083	One of the WI
kCameraDevicePropertyExposureIndexString	"Exposure Index"	One of the WI
kCameraDevicePropertyExposureMeteringMode	2084	One of the WI
kCameraDevicePropertyExposureMeteringModeString	"Exposure Metering Mode"	One of the WI
kCameraDevicePropertyExposureMode	2052	One of the WI
kCameraDevicePropertyExposureModeString	"Exposure Mode"	One of the WI
kCameraDevicePropertyExposureTime	2054	One of the WI
kCameraDevicePropertyExposureTimeString	"Exposure Time"	One of the WI
kCameraDevicePropertyFlashMode	2056	One of the WI
kCameraDevicePropertyFlashModeString	"Flash Mode"	One of the WI
kCameraDevicePropertyFnumber	2055	One of the WI
kCameraDevicePropertyFnumberString	"F Number"	One of the WI
kCameraDevicePropertyFocalLength	2086	One of the WI
kCameraDevicePropertyFocalLengthString	"Focus Length"	One of the WI
kCameraDevicePropertyFocusDistance	2085	One of the WI
kCameraDevicePropertyFocusDistanceString	"Focus Distance"	One of the WI
kCameraDevicePropertyFocusManualDist	2058	One of the WI
kCameraDevicePropertyFocusManualDistString	"Focus Manual Dist"	One of the WI
kCameraDevicePropertyFocusMeteringMode	2072	One of the WI
kCameraDevicePropertyFocusMeteringModeString	"Focus Metering Mode"	One of the WI
kCameraDevicePropertyFocusMode	2057	One of the WI
kCameraDevicePropertyFocusModeString	"Focus Mode"	One of the WI
kCameraDevicePropertyPanPosition	2060	One of the WI
kCameraDevicePropertyPanPositionString	"Pan Position"	One of the WI
kCameraDevicePropertyPictHeight	2069	One of the WI
kCameraDevicePropertyPictHeightString	"Picture Height"	One of the WI
kCameraDevicePropertyPicturesRemaining	2051	One of the WI
kCameraDevicePropertyPicturesRemainingString	"Pictures Remaining"	One of the WI
kCameraDevicePropertyPicturesTaken	2050	One of the WI
kCameraDevicePropertyPicturesTakenString	"Pictures Taken"	One of the WI
kCameraDevicePropertyPictWidth	2068	One of the WI
kCameraDevicePropertyPictWidthString	"Picture Width"	One of the WI

20.18 class WIAStreamMBS

20.18.1 class WIAStreamMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIAStreamMBS interface lets you read and write data to stream objects.
Example:

dim ItemName as string = "Hello.jpg"
dim f as FolderItem = SpecialFolder.Desktop.Child(ItemName)
dim s as new WIAStreamMBS(WIAStreamMBS.kModeWrite + WIAStreamMBS.kModeCreate, f)

Notes: Stream objects contain the data in a structured storage object, where storages provide the structure. Simple data can be written directly to a stream but, most frequently, streams are elements nested within a storage object. They are similar to standard files.

20.18.2 Methods

20.18.3 Clone as WIAStreamMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Clone method creates a new stream object with its own seek pointer that references the same bytes as the original stream.
Notes: The Clone method creates a new stream object for accessing the same bytes but using a separate seek pointer. The new stream object sees the same data as the source-stream object. Changes written to one object are immediately visible in the other. Range locking is shared between the stream objects.

The initial setting of the seek pointer in the cloned stream instance is the same as the current setting of the seek pointer in the original stream at the time of the clone operation.

Lasterror is set.

20.18.4 Commit(flags as Integer)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Commit method ensures that any changes made to a stream object open in transacted mode are reflected in the parent storage.
Notes: If the stream object is open in direct mode, Commit has no effect other than flushing all memory

buffers to the next-level storage object. The COM compound file implementation of streams does not support opening streams in transacted mode.
Lasterror is set.

Possible flags: kCommitConsolidate, kCommitDangerouslyCommitMerelyToDiskCache, kCommitDefault, kCommitOnlyIfCurrent and kCommitOverwrite.

20.18.5   Constructor(mode as Integer, file as folderitem)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens or creates a file and retrieves a stream to read or write to that file.
Example:

dim ItemName as string = "Hello.jpg"
dim f as FolderItem = SpecialFolder.Desktop.Child(ItemName)
dim s as new WIAStreamMBS(WIAStreamMBS.kModeWrite + WIAStreamMBS.kModeCreate, f)

Notes: mode: the flags. Use the kMode* constants.
path: the file path.

Lasterror is set.
See also:

20.18.6   Constructor(mode as Integer, path as string)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens or creates a file and retrieves a stream to read or write to that file.
Notes: mode: the flags. Use the kMode* constants.
path: the file path.

Lasterror is set.
See also:

20.18.7   CopyTo(other as WIAStreamMBS, length as UInt64)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CopyTo method copies a specified number of bytes from the current seek pointer in the stream to the current seek pointer in another stream.
Notes: other: The destination stream. The stream pointed to by pstm can be a new stream or a clone of the source stream.
length: The number of bytes to copy from the source stream.
ReadSize: Optional, a variable where the actual number of bytes read from the source.
WriteSize: Optional, a variable where the actual number of bytes written to the destination.

Lasterror is set.

The CopyTo method copies the specified bytes from one stream to another. It can also be used to copy a stream to itself. The seek pointer in each stream instance is adjusted for the number of bytes read or written. This method is equivalent to reading cb bytes into memory using Read and then immediately writing them to the destination stream using Write, although CopyTo will be more efficient.

The destination stream can be a clone of the source stream created by calling the Clone method.

If CopyTo returns an error, you cannot assume that the seek pointers are valid for either the source or destination. Additionally, the values of pcbRead and pcbWritten are not meaningful even though they are returned.

If CopyTo returns successfully, the actual number of bytes read and written are the same.

To copy the remainder of the source from the current seek pointer, specify the maximum large integer value for the cb parameter. If the seek pointer is the beginning of the stream, this operation copies the entire stream.
See also:

20.18.8 CopyTo(other as WIAStreamMBS, length as UInt64, byref ReadSize as UInt64, byref WriteSize as UInt64)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CopyTo method copies a specified number of bytes from the current seek pointer in the stream to the current seek pointer in another stream.
Notes: other: The destination stream. The stream pointed to by pstm can be a new stream or a clone of the source stream.
length: The number of bytes to copy from the source stream.
ReadSize: Optional, a variable where the actual number of bytes read from the source.
WriteSize: Optional, a variable where the actual number of bytes written to the destination.

Lasterror is set.

The CopyTo method copies the specified bytes from one stream to another. It can also be used to copy a stream to itself. The seek pointer in each stream instance is adjusted for the number of bytes read or written. This method is equivalent to reading cb bytes into memory using Read and then immediately writing them to the destination stream using Write, although CopyTo will be more efficient.

The destination stream can be a clone of the source stream created by calling the Clone method.

If CopyTo returns an error, you cannot assume that the seek pointers are valid for either the source or destination. Additionally, the values of pcbRead and pcbWritten are not meaningful even though they are returned.

If CopyTo returns successfully, the actual number of bytes read and written are the same.

To copy the remainder of the source from the current seek pointer, specify the maximum large integer value for the cb parameter. If the seek pointer is the beginning of the stream, this operation copies the entire stream.
See also:

20.18.9   Revert

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Revert method discards all changes that have been made to a transacted stream since the last Commit call.
Notes: On streams open in direct mode and streams using the COM compound file implementation of Revert, this method has no effect.
Lasterror is set.

20.18.10   Seek(value as Int64, Origin as Integer) as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Seek method changes the seek pointer to a new location.
Notes: The new location is relative to either the beginning of the stream, the end of the stream, or the current seek pointer.

value: The displacement to be added to the location indicated by the dwOrigin parameter. If dwOrigin is kSeekSet, this is interpreted as an unsigned value rather than a signed value.

Origin: The origin for the displacement specified in value. The origin can be the beginning of the file (kSeek-Set), the current seek pointer (kSeekCur), or the end of the file (kSeekEnd). For more information about values, see the kSeek* constants.

Retruns the new seek pointer from the beginning of the stream.

Seek changes the seek pointer so that subsequent read and write operations can be performed at a different location in the stream object. It is an error to seek before the beginning of the stream. It is not, however, an error to seek past the end of the stream. Seeking past the end of the stream is useful for subsequent write operations, as the stream byte range will be extended to the new seek position immediately before the write is complete.

You can also use this method to obtain the current value of the seek pointer by calling this method with the Origin parameter set to kSeekCur and the value parameter set to 0 so that the seek pointer is not changed. The current seek pointer is returned.

Lasterror is set.

20.18.11   SetSize(size as UInt64)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetSize method changes the size of the stream object.
Notes: SetSize changes the size of the stream object. Call this method to preallocate space for the stream. If the size parameter is larger than the current stream size, the stream is extended to the indicated size by filling the intervening space with bytes of undefined value. This operation is similar to the Write method if the seek pointer is past the current end of stream.

If the size parameter is smaller than the current stream, the stream is truncated to the indicated size.

The seek pointer is not affected by the change in stream size.

Calling SetSize can be an effective way to obtain a large chunk of contiguous space.

Lasterror is set.

20.18.12   Properties

20.18.13   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.18.14   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.18.15   Constants

Constants

Constant	Value	Description
kCommitConsolidate	8	One of the flags for commit.
		Windows 2000 and Windows XP: Indicates that a sto dated after it is committed, resulting in a smaller file on only on the outermost storage object that has been open It is not valid for streams. The kCommitConsolidate flag any other kCommit* flags.
kCommitDangerouslyCommitMerelyToDiskCache	4	One of the flags for commit.
		Commits the changes to a write-behind disk cache, but to the disk. In a write-behind disk cache, the operati actually writes to a disk cache, thus increasing perfor eventually written to the disk, but usually not until aft has already returned. The performance increase come increased risk of losing data if a problem occurs before the data in the cache is lost.
		If you do not specify this value, then committing chang objects is robust even if a disk cache is used. The two ensures that data is stored on the disk and not just to t
kCommitDefault	0	One of the flags for commit.
		You can specify this condition with kCommitConsolidat of the other three flags in this list of elements. Use thi readability of code.
kCommitOnlyIfCurrent	2	One of the flags for commit.
		Prevents multiple users of a storage object from ov changes. The commit operation occurs only if there to the saved storage object because the user most rece the saved version of the storage object is the same ver been editing. If other users have changed the storage eration fails and returns the STG_E_NOTCURRENT behavior, call the Commit method again using the kCo
kCommitOverwrite	1	One of the flags for commit.
		The commit operation can overwrite existing data to r quirements. This value is not recommended for typic not as robust as the default value. In this case, it is p operation to fail after the old data is overwritten, but completely committed. Then, neither the old version the storage object will be intact.
		You can use this value in the following cases:
		* The user is willing to risk losing the data.
		* The low-memory save sequence will be used to safely to a smaller file.
		* A previous commit returned STG_E_MEDIUMFUL existing data would provide enough space to commit object.
		Be aware that the commit operation verifies that adequ any overwriting occurs. Thus, even with this value s operation fails due to space requirements, the old data however, for data loss to occur with the kCommitOver the commit operation fails for any reason other than la
kModeConvert	&h20000	One of the mode constants for the file stream.
		Creates the new object while preserving existing data in tents". In the case of a storage object or a byte array, th into a stream regardless of whether the existing file or by tains a layered storage object. This flag can only be use storage object. It cannot be used within a storage objec ateStream. It is also not valid to use this flag and the k flag simultaneously.
kModeCreate	&h1000	One of the mode constants for the file stream.
		Undicates that an existing storage object or stream sho

20.19   class WIATransferCallbackMBS

20.19.1   class WIATransferCallbackMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for transfering data from the device into your application.
Notes: This class is only used for WIA 2.x.

20.19.2   Properties

20.19.3   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.19.4   Events

20.19.5   GetNextStream(ItemName as string, FullItemName as string) as WIAStreamMBS

Plugin Version: 10.3, Platform: Windows, Targets: .

Function: Gets a new stream for the specified item.
Notes: ItemName: Specifies the name of the item to create stream for.
FullItemName: Specifies the full name of the item to create stream for.

Returns a new WIAStreamMBS object or nil for an error.

When this method is implemented by an image processing filter, the Windows Image Acquisition (WIA) 2.0 minidriver calls it during image acquisition to get the destination stream from the client.

A filter's WIATransferCallbackMBS.GetNextStream must delegate to the application's callback method. The filter uses the stream returned by the application callback's WIATransferCallbackMBS.GetNextStream implementation to create its own stream that it passes back to the WIA 2.0 service. The filtering is done when the filter's stream calls the IStream::Write method.

The filter's stream cannot make any assumptions on the number of bytes that are written to it on each write, since the unfiltered image data may come from the WIA 2.0 Preview Component rather than the driver.

The WIA 2.0 Preview Component always writes the whole unfiltered image data into the filter's stream only once, which means that the filter's stream has one source writing into it. If both the driver and the preview component write into the filter's stream, the filter's stream cannot assume, for example, that it will receive the full header the first time IStream::Write is called although its corresponding driver always writes the header data first in one write. Nor can it assume that a subsequent write contains exactly one scan line. So the filtering stream may have to count the number of bytes written to it to determine, for example, where the image data starts.

The image processing filter's WIATransferCallbackMBS.GetNextStream implementation should read the properties needed for its image processing from the item for which the image is being acquired. The filter does not read the properties directly from the pWiaItem2 passed into InitializeFilter. Instead the filter must call FindItemByName on this WIA 2.0 item to obtain the actual WIA 2.0 item. The reason for this is that the image being acquired may actually be a child item of pWiaItem2. For example, during a folder acquisition the filter uses pWiaItem2 to obtain pWiaItem2's child items in WIATransferCallbackMBS.GetNextStream (during a folder acquisition the driver returns the images represented by the child items of WiaItemMBS). The same is true when the WIA 2.0 Preview Component calls into the image processing filter passing a child WIA 2.0 item.

20.19.6   TransferCallback(w as WIATransferParamsMBS) as Integer

Plugin Version: 10.3, Platform: Windows, Targets: .

Function: Provides progress and other notifications during a transfer.
Notes: Returns an error value or zero on success.

20.20   class WIATransferMBS

20.20.1   class WIATransferMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WIATransferMBS class provides stream-based transfer of data.
Notes: This class is for WIA 2.x.

20.20.2   Methods

20.20.3   Cancel

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Cancels the current transfer operation.
Notes: Lasterror is set.

20.20.4   Download(TransferCallback as WIATransferCallbackMBS)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initiates a data download to the caller.
Notes: TransferCallback: The WIATransferCallbackMBS object to receive progress details and specify the destination.

If a folder is downloaded, then all the child items of that folder are also transferred. Each item is transferred in a separate stream.

Lasterror is set.

20.20.5   EnumerateFormatInfo as WIAFormatInfoEnumeratorMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an enumerator for the transfer formats that the Windows Image Acquisition (WIA) 2.0 device supports.
Notes: Lasterror is set.

20.20.6 Upload(Source as WIAStreamMBS, TransferCallback as WIATransfer-CallbackMBS)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initiates a data upload of a single item from the caller.
Notes: Source: Specifies a pointer to the Stream data.
TransferCallback: Specifies a pointer to the caller's WIATransferCallbackMBS interface.

Lasterror is set.

20.20.7 Properties

20.20.8 Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.20.9 Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.21   class WIATransferParamsMBS

20.21.1   class WIATransferParamsMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class you subclass to get progress of the image transfer.
Notes: The WiaTransferParams is transmitted to an application during a data transfer by the Windows Image Acquisition (WIA) run-time system to the WiaTransferCallbackMBS.TransferCallback method.

20.21.2   Properties

20.21.3   ErrorStatus as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The status, or error state, of the device set by the driver; for example, "warming up".
Notes: (Read and Write property)

20.21.4   Message as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the status of the data transfer.
Notes: See the kMessage* constants
(Read and Write property)

20.21.5   PercentComplete as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the progress of the data transfer as a percentage.
Example:

dim w as WIATransferParamsMBS // your parameters
Progressbar1.maximum = 100
Progressbar1.value = w.PercentComplete

Notes: (Read and Write property)

20.21.6 TransferredBytes as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the amount of data transferred.
Notes: (Read and Write property)

20.21.7 Constants

Constants

Constant	Value	Description
kMessageDeviceStatus	4	One of the message constants.
kMessageEndOfStream	2	One of the message constants.
kMessageEndOfTransfer	3	One of the message constants.
kMessageNewPage	5	One of the message constants.
kMessageStatus	1	One of the message constants.

20.22   class WIAVideoMBS

20.22.1   class WIAVideoMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IWiaVideo interface provides methods that allow an application that uses Windows Image Acquisition (WIA) services to acquire still images from a streaming video device.
Notes: Note WIA does not support video devices in Windows Server 2003, Windows Vista, and later. For those versions of the Windows, use DirectShow to acquire images from video.

20.22.2   Methods

20.22.3   Constructor

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Requires Windows XP or Windows Server 2003.

20.22.4   CreateVideoByDevNum(DeviceNumber as integer, win as DesktopWindow, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

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

Function: The CreateVideoByDevNum method creates a connection to a streaming video device with the device number obtained from a Directshow enumeration.
Notes: DeviceNumber: Specifies the video device's Directshow device number.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.
See also:

20.22.5 CreateVideoByDevNum(DeviceNumber as Integer, win as window, StretchToFit-Parent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The CreateVideoByDevNum method creates a connection to a streaming video device with the device number obtained from a Directshow enumeration.
Notes: DeviceNumber: Specifies the video device's Directshow device number.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.
See also:

- 20.22.4 CreateVideoByDevNum(DeviceNumber as integer, win as DesktopWindow, StretchToFitParent as boolean, AutoBeginPlayback as boolean) 692

- 20.22.6 CreateVideoByDevNum(DeviceNumber as Integer, WindowHandle as Integer, StretchToFit-Parent as boolean, AutoBeginPlayback as boolean) 693

20.22.6 CreateVideoByDevNum(DeviceNumber as Integer, WindowHandle as Integer, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateVideoByDevNum method creates a connection to a streaming video device with the device number obtained from a Directshow enumeration.
Notes: DeviceNumber: Specifies the video device's Directshow device number.
WindowHandle: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to

TRUE, the video display fills the window.
See also:

- 20.22.4 CreateVideoByDevNum(DeviceNumber as integer, win as DesktopWindow, StretchToFitParent as boolean, AutoBeginPlayback as boolean)   692

- 20.22.5 CreateVideoByDevNum(DeviceNumber as Integer, win as window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)   693

20.22.7   CreateVideoByName(FriendlyName as string, win as DesktopWindow, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

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

Function: The CreateVideoByName method creates a connection to a streaming video device with the friendly device name obtained from a Directshow enumeration.
Notes: FriendlyName: Specifies the video device's friendly name obtained from a Directshow device enumeration.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.
See also:

- 20.22.8 CreateVideoByName(FriendlyName as string, win as window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)   694

- 20.22.9 CreateVideoByName(FriendlyName as string, WindowHandle as Integer, StretchToFitParent as boolean, AutoBeginPlayback as boolean)   695

20.22.8   CreateVideoByName(FriendlyName as string, win as window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The CreateVideoByName method creates a connection to a streaming video device with the friendly device name obtained from a Directshow enumeration.
Notes: FriendlyName: Specifies the video device's friendly name obtained from a Directshow device enumeration.
win: Specifies the window in which to display the streaming video.

StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.
See also:

20.22.9 CreateVideoByName(FriendlyName as string, WindowHandle as Integer, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateVideoByName method creates a connection to a streaming video device with the friendly device name obtained from a Directshow enumeration.
Notes: FriendlyName: Specifies the video device's friendly name obtained from a Directshow device enumeration.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to TRUE to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.
See also:

20.22.10 CreateVideoByWiaDevID(WiaDeviceID as string, win as Desktop-Window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

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

Function: The CreateVideoByWiaDevID method creates a connection to a streaming video device from its DeviceID.
Notes: WiaDeviceID: Specifies the value of the video device's DeviceID property.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to true to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.

In order for the function to succeed, the ImagesDirectory property must be specified first.
See also:

- 20.22.11 CreateVideoByWiaDevID(WiaDeviceID as string, win as window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)   696

- 20.22.12 CreateVideoByWiaDevID(WiaDeviceID as string, WindowHandle as Integer, StretchToFit-Parent as boolean, AutoBeginPlayback as boolean)   697

20.22.11 CreateVideoByWiaDevID(WiaDeviceID as string, win as window, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop only.

Function: The CreateVideoByWiaDevID method creates a connection to a streaming video device from its DeviceID.
Notes: WiaDeviceID: Specifies the value of the video device's DeviceID property.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to true to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to

TRUE, the video display fills the window.

In order for the function to succeed, the ImagesDirectory property must be specified first.
See also:

20.22.12 CreateVideoByWiaDevID(WiaDeviceID as string, WindowHandle as Integer, StretchToFitParent as boolean, AutoBeginPlayback as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateVideoByWiaDevID method creates a connection to a streaming video device from its DeviceID.
Notes: WiaDeviceID: Specifies the value of the video device's DeviceID property.
win: Specifies the window in which to display the streaming video.
StretchToFitParent: Specifies whether the video display is stretched to fit the parent window. Set this parameter to true if the display should be stretched to fit the parent window; otherwise, set to false.
AutoBeginPlayback: Specifies whether the streaming video begins playback as soon as this method returns. Set this parameter to true to cause immediate playback; set it to false to require a call to Play before video playback begins.

Lasterror is set.
By default, the video is displayed in the video device's default resolution. If bStretchToFitParent is set to TRUE, the video display fills the window.

In order for the function to succeed, the ImagesDirectory property must be specified first.
See also:

20.22.13 CurrentState as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the state of the video stream.
Notes: See the kState* constants.

20.22.14   DestroyVideo

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DestroyVideo method shuts down the streaming video.
Notes: To restart video playback, the application must call one of the CreateVideo methods again.
Lasterror is set.

Call this method only after a successful call to CreateVideoByWiaDevID, CreateVideoByDevNum, or CreateVideoByName.

20.22.15   Pause

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Pause method pauses video playback.
Notes: Call this method only after a successful call to CreateVideoByWiaDevID, CreateVideoByDevNum, or CreateVideoByName.
Lasterror is set.

20.22.16   Play

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Begins playback of streaming video.
Notes: Call this method only after a successful call to CreateVideoByWiaDevID, CreateVideoByDevNum, or CreateVideoByName.
Lasterror is set.

20.22.17   ResizeVideo(StretchToFitParent as boolean)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ResizeVideo method resizes the video playback to the largest supported resolution that fits inside the parent window.
Notes: Call this method whenever the parent window is moved or resized.

By default, the video is displayed in a supported resolution smaller than the parent window. If bStretchToFitParent is set to true, the video display fills the window.

Lasterror is set.

20.22.18   TakePicture as folderitem

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The TakePicture method extracts a still image from the video stream, and saves the image as a JPEG file.
Notes: Returns the full path and filename of the JPEG file that this method creates.
Lasterror is set.

The path and directory where the image file is saved are specified by the ImagesDirectory or ImageFolder property.
See also:

20.22.19   TakePicture as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The TakePicture method extracts a still image from the video stream, and saves the image as a JPEG file.
Notes: Returns the full path and filename of the JPEG file that this method creates.
Lasterror is set.

The path and directory where the image file is saved are specified by the ImagesDirectory or ImageFolder property.
See also:

20.22.20   Properties

20.22.21   Handle as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

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

20.22.22   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

20.22.23   ImagesDirectory as string

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the full path and directory where images are stored when calling the TakePicture method.
Notes: Lasterror is set.
(Read and Write computed property)

20.22.24   ImagesFolder as folderitem

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the folder where TakePicture stores the images.
Notes: Lasterror is set.
(Read and Write computed property)

20.22.25   PreviewVisible as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The PreviewVisible property specifies whether the video playback is visible in its parent window.
Notes: This does not affect the state of the video.
Lasterror is set.
(Read and Write computed property)

20.22.26   Constants

Constants

Constant	Value	Description
kStateCreatingVideo	2	One of the constants for the video state.
		One of the WiaVideoMBS CreateVideo methods was called and WIA is in the process of creating the video stream.
kStateDestroyingVideo	6	One of the constants for the video state.
		The application called DestroyVideo method, and WIA is in the process of destroying the video stream.
kStateNoVideo	1	One of the constants for the video state.
		No video stream exists. Call CreateVideoByWiaDevID, CreateVideoByDevNum, or CreateVideoByName to create a video.
kStateVideoCreated	3	One of the constants for the video state.
		A video stream has been successfully created, but playback has not yet started.
kStateVideoPaused	5	One of the constants for the video state.
		A video stream has been successfully created, and the video is paused. The application can now call the TakePicture method.
kStateVideoPlaying	4	One of the constants for the video state.
		A video stream has been successfully created, and the video is playing. The application can now call the TakePicture method.

Chapter 21

Network

21.1 class WindowsDNSRecordAAAAMBS

21.1.1 class WindowsDNSRecordAAAAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS IPv6 (AAAA) record.
Notes: Be aware that IPv6 is not available on Windows XP and older.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.1.2 Methods

21.1.3 Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordAAAAMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.1.4   Properties

21.1.5   Address as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The address in a human readable format.
Notes: (Read and Write property)

21.1.6   RawAddress as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The raw address as a 16byte string.
Notes: (Read and Write property)

21.2 class WindowsDNSRecordAMBS

21.2.1 class WindowsDNSRecordAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS address (A) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.2.2 Methods

21.2.3 Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordAMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.2.4 Properties

21.2.5 Address as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IPv4 address as a string.
Notes: (Read and Write property)

21.2.6 IPAddress as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The raw address as an integer.
Notes: (Read and Write property)

21.3   class WindowsDNSRecordMBS

21.3.1   class WindowsDNSRecordMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for DNS query results on Windows.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.3.2   Methods

21.3.3   Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.3.4   Query(name as string, type as Integer, options as Integer = 0) as WindowsDNSRecordMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Performs a DNS query on Windows.
Notes: The DnsQuery function type is the generic query interface to the DNS namespace, and provides application developers with a DNS query resolution interface.

name: a string that represents the name of the owner of the record set that is queried.
type: A value that represents the RR DNS Record Type that is queried.
Options: A value that contains a combination of kDNSQuery* constants.

Returns the record chain on success.

Applications that call the DnsQuery function build a query using a fully-qualified DNS name and Resource Record (RR) type, and set query options depending on the type of service desired. When the kDNSQuery-Standard option is set, DNS uses the resolver cache, queries first with UDP, then retries with TCP if the response is truncated, and requests that the server to perform recursive resolution on behalf of the client to resolve the query.

Note When calling one of the DnsQuery function types, be aware that a DNS server may return multiple records in response to a query. A computer that is multihomed, for example, will receive multiple A records for the same IP address. The caller must use as many of the returned records as necessary.

Consider the following scenario, in which multiple returned records require additional activity on behalf of the application: A Query function call is made for a multihomed computer and the application finds that the address associated with the first A record is not responding. The application should then attempt to use other IP addresses specified in the (additional) A records returned from the Query function call.

21.3.5   Properties

21.3.6   A as WindowsDNSRecordAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The address record.
Notes: (Read and Write property)

21.3.7   AAAA as WindowsDNSRecordAAAAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IPv6 address record.
Notes: (Read and Write property)

21.3.8   AFSDB as WindowsDNSRecordMXMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.9   CharSet as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This value specifies the character set used in the associated function call.
Notes: (Read and Write property)

21.3.10   CNAME as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.11   DataLength as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The length, in bytes, of Data.
Notes: (Read and Write property)

21.3.12   HINFO as WindowsDNSRecordTXTMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.13   ISDN as WindowsDNSRecordTXTMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.14   MB as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.15   MD as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.16   MF as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.17   MG as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.18   MINFO as WindowsDNSRecordMInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.19   MR as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.20   MX as WindowsDNSRecordMXMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.21 Name as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A pointer to a string that represents the domain name of the record set.
Notes: (Read and Write property)

21.3.22 NextRecord as WindowsDNSRecordMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The next record in the answer.
Notes: (Read and Write property)

21.3.23 NS as WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.24 Null as WindowsDNSRecordNullMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.25 RawData as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: the RAW data of this record.
Notes: For debugging.
(Read and Write property)

21.3.26 RP as WindowsDNSRecordMInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.27   RT as WindowsDNSRecordMXMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.28   Section as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This value specifies the section of interest returned from the DnsQuery function call.
Notes: (Read and Write property)

21.3.29   SOA as WindowsDNSRecordSOAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.30   TTL as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DNS RR's Time To Live value (TTL), in seconds.
Notes: (Read and Write property)

21.3.31   TXT as WindowsDNSRecordTXTMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.32   Type as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DNS record type for this record.
Notes: If type is kDNSTypeA, the A property is filled.
(Read and Write property)

21.3.33   X25 as WindowsDNSRecordTXTMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data record.
Notes: (Read and Write property)

21.3.34   Constants

Constants

Constant	Value	Description
kCharSetAnsi	3	One of the constants for the CharSet property. The character set is ANSI.
kCharSetUnicode	1	One of the constants for the CharSet property. The character set is Unicode.
kCharSetUnknown	0	One of the constants for the CharSet property. The character set is unknown.
kCharSetUtf8	2	One of the constants for the CharSet property. The character set is UTF8.
kDNSClassAll	&h00ff	One of the DNS classes in native byte order (little endian on Win
kDNSClassAny	&h00ff	One of the DNS classes in native byte order (little endian on Win
kDNSClassCHAOS	&h0003	One of the DNS classes in native byte order (little endian on Win
kDNSClassCSNET	&h0002	One of the DNS classes in native byte order (little endian on Win
kDNSClassHESIOD	&h0004	One of the DNS classes in native byte order (little endian on Win
kDNSClassInternet	&h0001	One of the DNS classes in native byte order (little endian on Win
kDNSClassNone	&h00fe	One of the DNS classes in native byte order (little endian on Win
kDNSQueryAcceptTruncatedResponse	&h00000001	One of the possible flags for the Query function. Returns truncated results. Does not retry under TCP.
kDNSQueryBypassCache	&h00000008	One of the possible flags for the Query function. Bypasses the resolver cache on the lookup.
kDNSQueryDontResetTTLValues	&h00100000	One of the possible flags for the Query function. If set, and if the response contains multiple records, records are the TTL corresponding to the minimum value TTL from among When this option is set, "Do not change the TTL of individual rec returned record set is not modified.
kDNSQueryMulticastOnly	&h00000400	One of the possible flags for the Query function. Prevents the query from using DNS and uses only Local Link Mul Resolution (LLMNR). Windows Vista and Windows Server 2008 or later.: This value is
kDNSQueryNoHostsFile	&h00000040	One of the possible flags for the Query function. Prevents the DNS query from consulting the HOSTS file. Windows 2000 Server and Windows 2000 Professional: This value ported.
kDNSQueryNoLocalName	&h00000020	One of the possible flags for the Query function. Directs DNS to ignore the local name. Windows 2000 Server and Windows 2000 Professional: This value ported.
kDNSQueryNoMulticast	&h00000800	One of the possible flags for the Query function.
kDNSQueryNoNetBT	&h00000080	One of the possible flags for the Query function. Prevents the DNS query from using NetBT for resolution. Windows 2000 Server and Windows 2000 Professional: This value ported.
kDNSQueryNoRecursion	&h00000004	One of the possible flags for the Query function. Directs the DNS server to perform an iterative query (specifically DNS server not to perform recursive resolution to resolve the que
kDNSQueryNoWireQuery	&h00000010	One of the possible flags for the Query function. Directs DNS to perform a query on the local cache only.
kDNSQueryReserved	&hff000000	One of the possible flags for the Query function. Reserved.
kDNSQueryReturnMessage	&h00000200	One of the possible flags for the Query function. Directs DNS to return the entire DNS response message. Windows 2000 Server and Windows 2000 Professional: This value ported.
kDNSQueryStandard	&h00000000	One of the possible flags for the Query function. Standard query.
kDNSQueryTreatAsFQDN	&h00001000	One of the possible flags for the Query function. Prevents the DNS response from attaching suffixes to the submit a name resolution process
kDNSQueryUseTCPOnly	&h00000002	One of the possible flags for the Query function.

714 CHAPTER 21. NETWORK

21.4 class WindowsDNSRecordMInfoMBS

21.4.1 class WindowsDNSRecordMInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS mail information (MINFO) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.4.2 Methods

21.4.3 Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordMInfoMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.4.4 Properties

21.4.5 NameErrorsMailbox as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that represents the FQDN of the mailbox to receive error messages related to the mailing list.
Notes: (Read and Write property)

21.4.6 NameMailbox as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that represents the fully qualified domain name (FQDN) of the mailbox responsible for the mailing list or mailbox specified in the record's owner name.
Notes: (Read and Write property)

21.5 class WindowsDNSRecordMXMBS

21.5.1 class WindowsDNSRecordMXMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS mail exchanger (MX) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.5.2 Methods

21.5.3 Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordMXMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.5.4 Properties

21.5.5 NameExchange as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that represents the fully qualified domain name (FQDN) of the host willing to act as a mail exchange.
Notes: (Read and Write property)

21.5.6 Preference as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A preference given to this resource record among others of the same owner. Lower values are preferred.
Notes: (Read and Write property)

21.6   class WindowsDNSRecordNullMBS

21.6.1   class WindowsDNSRecordNullMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a NULL data DNS resource record
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.6.2   Methods

21.6.3   Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordNullMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.6.4   Properties

21.6.5   ByteCount as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of bytes represented in Data.
Notes: (Read and Write property)

21.6.6   Data as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The raw data as a string.
Notes: (Read and Write property)

21.7 class WindowsDNSRecordPTRMBS

21.7.1 class WindowsDNSRecordPTRMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS pointer (PTR) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.7.2 Methods

21.7.3 Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordPTRMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.7.4 Properties

21.7.5 NameHost as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The named host.
Notes: (Read and Write property)

21.8   class WindowsDNSRecordSOAMBS

21.8.1   class WindowsDNSRecordSOAMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS start of authority (SOA) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.8.2   Methods

21.8.3   Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordSOAMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.8.4   Properties

21.8.5   DefaultTTL as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The lower limit on the time, in seconds, that a DNS server or caching resolver are allowed to cache any resource records (RR) from the zone to which this record belongs.
Notes: (Read and Write property)

21.8.6   Expire as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time, in seconds, before an unresponsive zone is no longer authoritative.
Notes: (Read and Write property)

21.8.7 NameAdministrator as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that represents the name of the responsible party for the zone to which the record belongs.
Notes: (Read and Write property)

21.8.8 NamePrimaryServer as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that represents the name of the authoritative DNS server for the zone to which the record belongs.
Notes: (Read and Write property)

21.8.9 Refresh as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time, in seconds, before the zone containing this record should be refreshed.
Notes: (Read and Write property)

21.8.10 Retry as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time, in seconds, before retrying a failed refresh of the zone to which this record belongs.
Notes: (Read and Write property)

21.8.11 SerialNo as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The serial number of the SOA record.
Notes: (Read and Write property)

21.9   class WindowsDNSRecordTXTMBS

21.9.1   class WindowsDNSRecordTXTMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a DNS text (TXT) record.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.9.2   Methods

21.9.3   Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: This constructor makes sure you don't create useless WindowsDNSRecordTXTMBS objects by error. The only way to create an object is to use the root method to get the object hierarchy.
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.

21.9.4   Strings as String()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: An array of strings representing the descriptive text of the TXT resource record.

21.9.5   Properties

21.9.6   StringCount as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of strings represented in Strings array.
Notes: (Read and Write property)

21.10 class WindowsProxyMBS

21.10.1 class WindowsProxyMBS

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to query the current windows proxy settings.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.2pr5
- Need Proxy Settings?

21.10.2 Properties

21.10.3 ByPass as String

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The proxy
Notes: Example values:
"testserver:1234 apple.de <local>"
or
"testserver3:1234 testserver4:1234 testserver1:1234 testserver2:1234 apple.de <local>"
(Read and Write property)

21.10.4 Proxy as String

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current proxy setting.
Notes: Example values:
"testserver:1234"
or
"https=testserver2:1234 http=testserver1:1234 gopher=testserver4:1234 ftp=testserver3:1234"
(Read and Write property)

21.10.5 UsingProxy as Boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether a proxy is in use.
Notes: (Read and Write property)

21.11 class WindowsQOSMBS

21.11.1 class WindowsQOSMBS

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for Quality Windows Audio/Video Experience (qWAVE).
Notes: Quality Windows Audio/Video Experience (qWAVE) is the next generation Quality of Service (QOS) platform introduced in Windows Vista.
qWAVE provides new features focused on streaming multimedia and real-time content over variable bandwidth networks. These features include the following.

- Auto-discovery of end-to-end QOS compatibility.

- End-to-end bandwidth estimation of maximum link capacity (bottleneck bandwidth) and real-time available bandwidth.

- Intelligent packet prioritization.

- Congestion notification.

- Flow shaping.

- Distributed admission control, including caching to improve performance and minimize latency.

Requires Windows 6 (Vista).
Blog Entries

- MBS Xojo Plugins, version 21.4pr4

- MBS Xojo Plugins, version 21.2pr7

- MBS Xojo / Real Studio plug-ins in version 13.5

- MBS Xojo / Real Studio Plugins, version 13.5pr5

Xojo Developer Magazine

- 12.1, page 9: News

21.11.2 Methods

21.11.3 AddSocketToFlow(Socket as Integer, DestAddr as string, DestPort as Integer, TrafficType as Integer, Flags as Integer, byref FlowId as UInt32) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a socket to a flow.

Notes: Socket: Identifies the socket that the application will use to flow traffic.

DestAddr: The destination IP address to which the application will send traffic. DestAddr and DestPort are optional if the socket is already connected. If this parameter is specified, the remote IP address and port must match those used in the socket's connect call.

If the socket is not connected, this parameter must be specified. If the socket is already connected, this parameter does not need to be specified. In this case, if the parameter is still specified, the destination host and port must match what was specified during the socket connect call.

Since, under TCP, the socket connect call can be delayed, AddSocketToFlow can be called before a connection is established, passing in the remote system's IP address and port number in the DestAddr parameter.

DestPort: The port for the connection.

TrafficType: A QOSTrafficType* constant that specifies the type of traffic for which this flow will be used.

Flags: Optional flag values. Can be QOSNonAdaptiveFlow.

FlowID: Receives a flow identifier. On input, this value must be 0. On output, the buffer contains a flow identifier if the call succeeds.

If a socket is being added to an existing flow, this parameter will be the identifier of that flow.

An application can make use of this parameter if multiple sockets used can share the same QoS flow properties. The QoS subsystem, then does not have to incur the overhead of provisioning new flows for subsequent sockets with the same properties. Note that only non-adaptive flows can have multiple sockets attached to an existing flow.

A FlowID is an unsigned 32-bit integer.

Lasterror is set.

The use of IPv4/v6 mixed addresses is not supported in qWAVE. The address specified by the DestAddr parameter must be either IPv4 or IPv6.

If there is a requirement for network experiments over a specific network interface, the socket must be bound to that particular interface. Otherwise the most appropriate interface for the experiment, as indicated by the network stack, is assigned by the qWAVE subsystem.

Network traffic associated with this flow is not affected by making this call alone. For example, packet prioritization does not occur immediately.

There are two categories of applications that use this function: adaptive and non-adaptive. An adaptive application makes use of notifications and information in the FlowFundamentals for adapting to network changes such as congestion. The qWAVE service uses Link Layer Topology Discovery (LLTD) QoS extensions for adaptive flows which can be present on the destination device.

After calling this function adaptive A/V applications should call the SetFlowRate function to affect network traffic.

A non-adaptive application either does not adapt to changing network characteristics or is sending traffic to an endpoint that does not support adaptive capabilities as indicated by ERROR_NOT_SUPPORTED.

Non-adaptive applications, or adaptive applications making non-adaptive flows, should call this function with the QOSNonAdaptiveFlow flag. After calling this function A/V applications should call the setOutgoingDSCPValue, setOutgoingRate or setTrafficType function with a Operation. They do not need to be called unless shaping is desired.

See also:

21.11.4 AddSocketToFlow(Socket as Integer, TrafficType as Integer, Flags as Integer, byref FlowId as UInt32) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a socket to a flow.
Notes: Socket: Identifies the socket that the application will use to flow traffic.
DestAddr: The destination IP address to which the application will send traffic. DestAddr and DestPort are optional if the socket is already connected. If this parameter is specified, the remote IP address and port must match those used in the socket's connect call.
If the socket is not connected, this parameter must be specified. If the socket is already connected, this parameter does not need to be specified. In this case, if the parameter is still specified, the destination host and port must match what was specified during the socket connect call.
Since, under TCP, the socket connect call can be delayed, AddSocketToFlow can be called before a connection is established, passing in the remote system's IP address and port number in the DestAddr parameter.
DestPort: The port for the connection.
TrafficType: A QOSTrafficType* constant that specifies the type of traffic for which this flow will be used.
Flags: Optional flag values. Can be QOSNonAdaptiveFlow.
FlowID: Receives a flow identifier. On input, this value must be 0. On output, the buffer contains a flow identifier if the call succeeds.
If a socket is being added to an existing flow, this parameter will be the identifier of that flow.
An application can make use of this parameter if multiple sockets used can share the same QoS flow properties. The QoS subsystem, then does not have to incur the overhead of provisioning new flows for subsequent sockets with the same properties. Note that only non-adaptive flows can have multiple sockets attached to an existing flow.
A FlowID is an unsigned 32-bit integer.

Lasterror is set.

The use of IPv4/v6 mixed addresses is not supported in qWAVE. The address specified by the DestAddr parameter must be either IPv4 or IPv6.
If there is a requirement for network experiments over a specific network interface, the socket must be bound to that particular interface. Otherwise the most appropriate interface for the experiment, as indicated by the network stack, is assigned by the qWAVE subsystem.
Network traffic associated with this flow is not affected by making this call alone. For example, packet prioritization does not occur immediately.
There are two categories of applications that use this function: adaptive and non-adaptive. An adaptive application makes use of notifications and information in the FlowFundamentals for adapting to network changes such as congestion. The qWAVE service uses Link Layer Topology Discovery (LLTD) QoS extensions for adaptive flows which can be present on the destination device.
After calling this function adaptive A/V applications should call the SetFlowRate function to affect network traffic.
A non-adaptive application either does not adapt to changing network characteristics or is sending traffic to an endpoint that does not support adaptive capabilities as indicated by ERROR_NOT_SUPPORTED.
Non-adaptive applications, or adaptive applications making non-adaptive flows, should call this function with the QOSNonAdaptiveFlow flag. After calling this function A/V applications should call the setOutgoingDSCPValue, setOutgoingRate or setTrafficType function with a Operation. They do not need to be

called unless shaping is desired.
See also:

- 21.11.3 AddSocketToFlow(Socket as Integer, DestAddr as string, DestPort as Integer, TrafficType as Integer, Flags as Integer, byref FlowId as UInt32) as boolean 722

21.11.5 Constructor

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: This function initializes the QOS subsystem.
Notes: On success Handle property is not zero.
Lasterror is set.

Every process intending to use qWAVE must first call Constructor.
If a machine enters a power save mode that interrupts connectivity such as sleep or standby, existing and active network experiments such as QOSStartTrackingClient must be reinitiated. This recreation of the flow mirrors the cleanup and creation activities also necessary for existing sockets. A new handle must be created, and the flow must be recreated and readmitted.

21.11.6 getFlowFundamentals(FlowID as Integer, byref BottleneckBandwidthSet as boolean, byref BottleneckBandwidth as UInt64, byref AvailableBandwidthSet as boolean, byref AvailableBandwidth as UInt64, byref RTTSet as boolean, byref RTT as UInt32, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries flow fundamentals.
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.
Flags: Optional, can be QOSQueryFlowFresh.
BottleneckBandwidthSet: This Boolean value is set to true if the BottleneckBandwidth field contains a value.
BottleneckBandwidth: Indicates the maximum end-to-end link capacity between the source and sink device, in bits.
AvailableBandwidthSet: Set to true if the AvailableBandwidth field contains a value.
AvailableBandwidth: Indicates how much bandwidth is available for submitting traffic on the end-to-end network path between the source and sink device, in bits.
RTTSet: Set to true if the RTT field contains a value.
RTT: Measures the round-trip time between the source and sink device, in microseconds.

Lasterror is set. Returns true on success or false on failure.

21.11.7 getOutgoingRate(FlowID as Integer, byref Bandwidth as UInt64, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries flow rate.
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.
Bandwidth: will contain a UINT64 value that indicates the flow rate specified when requesting the contract, in bits per second.
Flags: Optional, can be QOSQueryFlowFresh.
Lasterror is set. Returns true on success or false on failure.

21.11.8 getPacketPriority(FlowID as Integer, byref ConformantDSCPValue as Integer, byref NonConformantDSCPValue as Integer, byref ConformantL2Value as Integer, byref NonConformantL2Value as Integer, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries packet priorities.
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.
Flags: Optional, can be QOSQueryFlowFresh.

ConformantDSCPValue: Differential Services Code Point (DSCP) mark used for flow traffic that conforms to the specified flow rate.
NonConformantDSCPValue: DSCP marking used for flow traffic that exceeds the specified flow rate. Non-conformant DSCP values are only applicable only if value is QOSUseNonConformantMarkings.
ConformantL2Value: Layer-2 (L2) tag used for flow traffic that conforms to the specified flow rate. L2 tags will not be added to packets if the end-to-end path between source and sink does not support them.
NonConformantL2Value: L2 tag used for flow traffic that exceeds the specified flow rate. Non-conformant L2 values are only applicable if value is QOSUseNonConformantMarkings.

Lasterror is set. Returns true on success or false on failure.

21.11.9 RemoveAllSocketsFromFlow(FlowID as Integer) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The RemoveSocketFromFlow function notifies the QOS subsystem that a previously added flow has been terminated by the application, and that the subsystem must update its internal information accordingly.
Notes: FlowId: A flow identifier. A QOS_FLOWID is an unsigned 32-bit integer.

Lasterror is set. Returns true on success or false on failure.
The plugin closes all flows in destructor.

21.11.10 RemoveSocketFromFlow(socketHandle as Integer, FlowID as Integer) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The RemoveSocketFromFlow function notifies the QOS subsystem that a previously added flow has been terminated by the application, and that the subsystem must update its internal information accordingly.
Notes: Socket: Socket to be removed from the flow.
Only flows created with the QOSNonAdaptiveFlow flag may have multiple sockets added to the same flow. By passing the Socket parameter in this call, each socket can be removed individually. If the Socket parameter is not passed, the entire flow will be destroyed. If only one socket was attached to the flow, passing this socket as a parameter to this function and passing 0 as a socket are equivalent calls.
FlowId: A flow identifier. A QOS_FLOWID is an unsigned 32-bit integer.

Lasterror is set. Returns true on success or false on failure.
The plugin closes all flows in destructor.

21.11.11 setOutgoingDSCPValue(FlowID as Integer, OutgoingDSCPValue as Integer, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the outgoing DSCP value.
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.

Lasterror is set. Returns true on success or false on failure.

If StartTrackingClient has not already been called, calling QOSSetFlow will cause the QOS subsystem to perform the following.

- Discover whether the end-to-end network path supports prioritization.

- Track end-to-end network characteristics by way of network experiments. These experiments do not place any noteworthy stress on the network.

21.11.12   setOutgoingRate(FlowID as Integer, Bandwidth as UInt64, Shaping-Behavior as Integer, Reason as Integer, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets a new outgoing rate.
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.
Bandwidth: The rate at which data should be sent, in units of bits per second.
Note Traffic on the network is measured at the IP level, and not at the application level. The rate that is specified should account for the IP and protocol headers.
ShapingBehavior: A shaping constant that defines the shaping behavior of the flow.
Reason: A QOSFlowRate* constant that indicates the reason for a flow rate change.
Lasterror is set. Returns true on success or false on failure.

If StartTrackingClient has not already been called, calling QOSSetFlow will cause the QOS subsystem to perform the following.

- Discover whether the end-to-end network path supports prioritization.

- Track end-to-end network characteristics by way of network experiments. These experiments do not place any noteworthy stress on the network.#

21.11.13   setTrafficType(FlowID as Integer, TrafficType as Integer, Flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the type of traffic
Notes: FlowID: A flow identifier. A flow id is an unsigned 32-bit integer.
TrafficType: One of the traffic type constants.
Lasterror is set. Returns true on success or false on failure.

If StartTrackingClient has not already been called, calling QOSSetFlow will cause the QOS subsystem to perform the following.

- Discover whether the end-to-end network path supports prioritization.

- Track end-to-end network characteristics by way of network experiments. These experiments do not place any noteworthy stress on the network.#

21.11.14   StartTrackingClient(DestAddr as string, flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StartTrackingClient function notifies the QOS subsystem of the existence of a new client.
Notes: Calling this function increases the likelihood that the QOS subsystem will have gathered sufficient information on the network path to assist when calling QOSSetFlow to set the flow.
Note This call is not required to add a flow with the QOSAddSocketToFlow function although it is highly recommended. Not calling this function may require network experiments to be started during the QOSSet-Flow call and can result in QOSSetFlow failing with ERROR_NETWORK_BUSY on initial use.

DestAddr: The IP address of the client device. Clients are identified by their IP address and address family. Any port number specified in the sockaddr structure will be ignored.
Flags: Reserved for future use. Must be set to 0.

On receipt of a QOSStartTrackingClient call the QoS subsystem begins gathering information about the client such as the QoS capabilities and available bandwidth on the end-to-end path.
An application should call this function as soon as it becomes aware of a client device that may need QoS flow. For example this function should be called when a media player device first connects to a media server application.
Network experiments performed by QOSStartTrackingClient do not introduce noteworthy load on the network even if no stream is started for a long period of time. The qWAVE service dynamically adjusts experiment traffic based on QoS subsystem activity.
Link Layer Topology Discovery (LLTD) must be implemented on the sink PC or device for this function to work.

Lasterror is set. Returns true on success or false on failure.

21.11.15   StopTrackingClient(DestAddr as string, flags as Integer = 0) as boolean

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StopTrackingClient function notifies the QoS subsystem to stop tracking a client that has previously used the StartTrackingClient function.
Notes: If a flow is currently in progress, this function will not affect it.

DestAddr: The IP address of the client device. Clients are identified by their IP address and address family. A port number is not required and will be ignored.
Flags: Reserved for future use.

Lasterror is set. Returns true on success or false on failure.

21.11.16   Properties

21.11.17   Handle as Integer

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

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

21.11.18   Lasterror as Integer

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

21.11.19   LasterrorMessage as String

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The generic window error message for the error in lasterror property.
Notes: (Read and Write property)

21.11.20   Constants

Flow Reasons

Constant	Value	Description
QOSFlowRateCongestion	2	Indicates that the flow has changed due to congestion.
QOSFlowRateContentChange	1	Indicates that the content of a flow has changed.
QOSFlowRateHigherContentEncoding	3	Indicates that the user has caused the flow to change.
QOSFlowRateNotApplicable	0	Indicates that there has not been a change in the flow.
QOSFlowRateUserCaused	4	Indicates that the user has caused the flow to change.

Flags

Constant	Value	Description
QOSNonAdaptiveFlow	2	If specified, the QoS subsystem will not gather data about the network path for this flow. As a result, functions which rely on bandwidth estimation techniques will not be available. For example, this would block QOSQueryFlow with an Operation value of QOSQueryFlowFundamentals and QOSNotifyFlow with an Operation value of QOSNotifyCongested, QOSNotifyUncongested, and QOSNotifyAvailable.
QOSQueryFlowFresh	1	The QOS subsystem will only return fresh, not cached, data. If fresh data is unavailable, it will try to obtain such data, at the expense of possibly taking more time. If this is not possible, the call will fail with the error code ERROR_RETRY. This flag is only applicable when the Operation parameter is set to QOSQueryFlowFundamentals.

Shaping Behavior

Constant	Value	Description
QOSShapeAndMark	1	Indicates that the Windows Scheduler will be used to enforce the requested flow rate. Data packets exceeding the rate are delayed accordingly. Packets receive conformant priority values.
QOSShapeOnly	0	Indicates that the Windows packet scheduler (Pacer) will be used to enforce the requested flow rate. Data packets that exceed the rate are delayed until appropriate in order to maintain the specified flow rate. If the network supports prioritization, packets will always receive conformant priority values when QOSShapeFlow is specified.
QOSUseNonConformantMarkings	2	Indicates that the flow rate requested will not be enforced. Data packets that would exceed the flow rate will receive a priority that indicates they are non conformant. This may lead to lost and reordered packets.

Traffic Types

Constant	Value	Description
QOSTrafficTypeAudioVideo	3	Flow traffic has a network priority higher than QOSTrafficTypeExcellentEffort, yet lower than QOSTrafficTypeVoice. This traffic type should be used for A/V streaming scenarios such as MPEG2 streaming. Sent traffic will contain a DSCP mark with a value of 0x28 and an 802.1p tag with a value of 5.
QOSTrafficTypeBackground	1	Flow traffic has a network priority lower than that of QOSTrafficTypeBestEffort. This traffic type could be used for traffic of an application doing data backup. Sent traffic will contain a DSCP mark with a value of 0x08 and an 802.1p tag with a value of 2.
QOSTrafficTypeBestEffort	0	Flow traffic has the same network priority as regular traffic not associated with QOS. This traffic type is the same as not specifying priority, and as a result, the DSCP mark and 802.1p tag are not added to sent traffic.
QOSTrafficTypeControl	5	Flow traffic has the highest network priority. This traffic type should only be used for the most critical of data. For example, it may be used for data carrying user inputs. Sent traffic will contain a DSCP mark with a value of 0x38 and an 802.1p tag with a value of 7.
QOSTrafficTypeExcellentEffort	2	Flow traffic has a network priority higher than QOSTrafficTypeBestEffort, yet lower than QOSTrafficTypeAudioVideo. This traffic type should be used for data traffic that is more important than normal end-user scenarios, such as email. Sent traffic will contain a DSCP mark with value of 0x28 and 802.1p tag with a value of 5.
QOSTrafficTypeVoice	4	Flow traffic has a network priority higher than QOSTrafficTypeAudioVideo, yet lower than QOSTrafficTypeControl. This traffic type should be used for realtime voice streams such as VOIP. Sent traffic will contain a DSCP mark with a value of 0x38 and an 802.1p tag with a value of 7.

21.12 class WinHTTPClientAutoProxyOptionsMBS

21.12.1 class WinHTTPClientAutoProxyOptionsMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WinHTTPClientAutoProxyOptionsMBS class is used to indicate to the WinHttpGetProxy-ForURL function whether to specify the URL of the Proxy Auto-Configuration (PAC) file or to automatically locate the URL with DHCP or DNS queries to the network.
Notes: see also:
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384123(v=vs.85).aspx

21.12.2 Methods

21.12.3 Constructor

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

21.12.4 Properties

21.12.5 AutoConfigUrl as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The URL for the config URL.
Notes: If Flags includes the kAutoProxyConfigURL flag, the AutoConfigUrl contains the URL of the proxy auto-configuration (PAC) file.
If Flags does not include the kAutoProxyConfigURL flag, then AutoConfigUrl must be "".
(Read and Write property)

21.12.6 AutoDetectFlags as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The flags for auto detecting.
Notes: If Flags includes the kAutoProxyAutoDetect flag, then AutoDetectFlags specifies what protocols are to be used to locate the PAC file. If both the DHCP and DNS auto detect flags are specified, then DHCP is used first; if no PAC URL is discovered using DHCP, then DNS is used.
If Flags does not include the kAutoProxyAutoDetect flag, then AutoDetectFlags must be zero.

Value	Meaning
kAutoDetectTypeDHCP	Use DHCP to locate the proxy auto-configuration file.
kAutoDetectTypeDNSA	Use DNS to attempt to locate the proxy auto-configuration file at a well-known location on the domain of the local computer.

(Read and Write property)

21.12.7   AutoLogonIfChallenged as Boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies whether the client's domain credentials should be automatically sent in response to an NTLM or Negotiate Authentication challenge when WinHTTP requests the PAC file.
Notes: If this flag is true, credentials should automatically be sent in response to an authentication challenge. If this flag is false and authentication is required to download the PAC file, the WinHttpGetProxyForUrl function fails.
(Read and Write property)

21.12.8   Flags as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Flags for detecting proxy.
Notes: (Read and Write property)

21.12.9   Constants

Auto Detect Types

Constant	Value	Description
kAutoDetectTypeDHCP	1	Use DHCP to locate the proxy auto-configuration file.
kAutoDetectTypeDNSA	2	Use DNS to attempt to locate the proxy auto-configuration file at a well-known location on the domain of the local computer.

Proxy Flags

Constant	Value	Description
kAutoProxyAutoDetect	1	Attempt to automatically discover the URL of the PAC file using both DHCP and DNS queries to the local network.
kAutoProxyConfigURL	2	Download the PAC file from the URL specified by AutoConfigUrl in the WinHTTPClientAutoProxyOptionsMBS class.

21.13   class WinHTTPClientCurrentUserIEProxyConfigMBS

21.13.1   class WinHTTPClientCurrentUserIEProxyConfigMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for Internet Explorer proxy configuration information.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384250(v=vs.85).aspx
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.13.2   Methods

21.13.3   Constructor

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

21.13.4   Properties

21.13.5   AutoConfigUrl as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The auto-configuration URL.
Notes: The auto-configuration URL if the Internet Explorer proxy configuration for the current user specifies "Use automatic proxy configuration".
(Read and Write property)

21.13.6   AutoDetect as Boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to auto detect proxy settings.
Notes: If true, indicates that the Internet Explorer proxy configuration for the current user specifies "automatically detect settings".
(Read and Write property)

21.13.7 Proxy as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Contains the proxy URL if the Internet Explorer proxy configuration for the current user specifies "use a proxy server".
Notes: (Read and Write property)

21.13.8 ProxyBypass as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Optional proxy by-pass server list.
Notes: (Read and Write property)

21.14 class WinHTTPClientMBS

21.14.1 class WinHTTPClientMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for Windows to do http requests.
Notes: The plugin only implements proxy related functions.
But this class could be extended to work like HTTPSocket.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.1

- MBS Xojo / Real Studio Plugins, version 14.1pr1

21.14.2 Methods

21.14.3 Close as boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Closes the session.

21.14.4 Constructor

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: The constructor.

21.14.5 CrackUrl(URL as string, Flags as Integer = 0) as WinHTTPClientURLComponentsMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Separates a URL into its component parts such as host name and path.
Notes: URL: the canonical URL to separate. WinHttpCrackUrl does not check this URL for validity or correct format before attempting to crack it.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384092(v=vs.85).aspx

21.14.6 DetectAutoProxyConfigUrl(AutoDetectFlags as Integer, byref Auto-ConfigUrl as string) as Boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Finds the URL for the Proxy Auto-Configuration (PAC) file.
Notes: This function reports the URL of the PAC file, but it does not download the file.

AutoDetectFlags: A data type that specifies what protocols to use to locate the PAC file. If both the DHCP and DNS auto detect flags are set, DHCP is used first; if no PAC URL is discovered using DHCP, then DNS is used.

Value	Meaning
kAutoDetectTypeDHCP	Use DHCP to locate the proxy auto-configuration file.
kAutoDetectTypeDNSA	Use DNS to attempt to locate the proxy auto-configuration file at a well-known location on the domain of the local computer.

AutoConfigUrl: String that contains the configuration URL that receives the proxy data.

Returns true if successful, or false otherwise. For extended error information, call LastError. Among the error codes returned are the following.

WinHTTP implements the Web Proxy Auto-Discovery (WPAD) protocol, often referred to as autoproxy. For more information about well-known locations, see the Discovery Process section of the WPAD protocol document.
Note that because the DetectAutoProxyConfigUrl function takes time to complete its operation, it should not be called from a UI thread.

21.14.7 GetDefaultProxyConfiguration as WinHTTPClientProxyInfoMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Retrieves the default WinHTTP proxy configuration from the registry.
Notes: Returns the default proxy configuration.

21.14.8 GetIEProxyConfigForCurrentUser as WinHTTPClientCurrentUserIEProxyConfigMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Retrieves the Internet Explorer proxy configuration for the current user.

Notes: Queries the Internet Explorer proxy settings for the current active network connection (for example, LAN, dial-up, or VPN connection).

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384096(v=vs.85).aspx

21.14.9 GetProxyForHost(URL as string, Host as string, byref proxy as string, byref proxyPort as string, AutoConfigURL as string = "") as boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Queries the proxy for a host.
Notes: This is an universal functions using InternetGetProxyInfo, DetectAutoProxyUrl and InternetQueryOption.

URL: The target URL.
Host: The hostname of the target server.

Returns true on success and sets proxy and port. Returns false on failure.
AutoConfigURL is only used on first call instead of a configured auto config URL in Internet Explorer.

21.14.10 GetProxyForUrl(URL as string, AutoProxyOptions as WinHTTPClientAutoProxyOptionsMBS, byref ProxyInfo as WinHTTPClientProxyInfoMBS) as boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Retrieves the proxy data for the specified URL.
Notes: Url: Contains the URL of the HTTP request that the application is preparing to send.
AutoProxyOptions: Specifies the auto-proxy options to use.
ProxyInfo: Receives the proxy setting.

If the function succeeds, the function returns true.
If the function fails, it returns false. Lasterror is set.

This function implements the Web Proxy Auto-Discovery (WPAD) protocol for automatically configuring the proxy settings for an HTTP request. The WPAD protocol downloads a Proxy Auto-Configuration (PAC) file, which is a script that identifies the proxy server to use for a given target URL. PAC files are typically deployed by the IT department within a corporate network environment. The URL of the PAC file can either be specified explicitly or GetProxyForUrl can be instructed to automatically discover the location of the PAC file on the local network.

GetProxyForUrl supports only ECMAScript-based PAC files.
GetProxyForUrl must be called on a per-URL basis, because the PAC file can return a different proxy server
for different URLs. This is useful because the PAC file enables an IT department to implement proxy server
load balancing by mapping (hashing) the target URL (specified by the lpcwszUrl parameter) to a certain
proxy in a proxy server array.
GetProxyForUrl caches the autoproxy URL and the autoproxy script when auto-discovery is specified in the
dwFlags member of the pAutoProxyOptions structure. For more information, see Autoproxy Cache.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384097(v=vs.85).aspx

21.14.11   InternetGetProxyInfo(URL as string, Host as string) as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Retrieves proxy data for accessing specified resources.
Notes: This function can only be called by dynamically linking to "JSProxy.dll" (plugin loads this for you).
For better autoproxy support, use HTTP Services (WinHTTP) version 5.1 instead.

Url: Secifies the URL of the target HTTP resource.
HostName: Specifies the host name of the target URL.

Returns the URL of the proxy to use in an HTTP request for the specified resource.
Returns empty string on any error.

21.14.12   Open(UserAgent as string, AccessType as Integer, ProxyName as string = "", ProxyByPass as string = "") as boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Initializes an application's use of the WinINet functions.
Notes: lUserAgent: Specifies the name of the application or entity calling the WinINet functions. This
name is used as the user agent in the HTTP protocol.
AccessType: Type of access required. This parameter can be one of the following values.

ProxyName: String that specifies the name of the proxy server(s) to use when proxy access is specified
by setting dwAccessType to INTERNET_OPEN_TYPE_PROXY. Do not use an empty string, because
InternetOpen will use it as the proxy name. The WinINet functions recognize only CERN type proxies
(HTTP only) and the TIS FTP gateway (FTP only). If Microsoft Internet Explorer is installed, these
functions also support SOCKS proxies. FTP requests can be made through a CERN type proxy either by

Value	Meaning
INTERNET_OPEN_TYPE_DIRECT	Resolves all host names locally.
INTERNET_OPEN_TYPE_PRECONFIG	Retrieves the proxy or direct configuration from the registry.
INTERNET_OPEN_TYPE_PRECONFIG_WITH_NO_AUTOPROXY	Retrieves the proxy or direct configuration from the registry and prevents the use of a startup Microsoft JScript or Internet Setup (INS) file.
INTERNET_OPEN_TYPE_PROXY	Passes requests to the proxy unless a proxy bypass list is supplied and the name to be resolved bypasses the proxy. In this case, the function uses INTERNET_OPEN_TYPE_DIRECT.

changing them to an HTTP request or by using InternetOpenUrl. If dwAccessType is not set to INTERNET_OPEN_TYPE_PROXY, this parameter is ignored and should be NULL. For more information about listing proxy servers, see the Listing Proxy Servers section of Enabling Internet Functionality.

ProxyBypass: Specifies an optional list of host names or IP addresses, or both, that should not be routed through the proxy when dwAccessType is set to INTERNET_OPEN_TYPE_PROXY. The list can contain wildcards. Do not use an empty string, because InternetOpen will use it as the proxy bypass list. If this parameter specifies the ”<local>” macro, the function bypasses the proxy for any host name that does not contain a period.

By default, WinINet will bypass the proxy for requests that use the host names ”localhost”, ”loopback”, ”127.0.0.1”, or ” [::1] ”. This behavior exists because a remote proxy server typically will not resolve these addresses properly.

Internet Explorer 9: You can remove the local computer from the proxy bypass list using the ”<-loopback>” macro.

If dwAccessType is not set to INTERNET_OPEN_TYPE_PROXY, this parameter is ignored and should be NULL.

Flags: Options. This parameter can be a combination of the following values.

Value	Meaning
INTERNET_FLAG_ASYNC	Makes only asynchronous requests on handles descended from the handle returned from this function.
INTERNET_FLAG_FROM_CACHE	Does not make network requests. All entities are returned from the cache. If the requested item is not in the cache, a suitable error, such as ERROR_FILE_NOT_FOUND, is returned.
INTERNET_FLAG_OFFLINE	Identical to INTERNET_FLAG_FROM_CACHE. Does not make network requests. All entities are returned from the cache. If the requested item is not in the cache, a suitable error, such as ERROR_FILE_NOT_FOUND, is returned.

Returns a valid handle that the application passes to subsequent WinINet functions. If Open fails, it returns false. To retrieve a specific error message, call GetLastError.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa385096(v=vs.85).aspx

21.14.13 SetDefaultProxyConfiguration(info as WinHTTPClientProxyInfoMBS) as boolean

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: Sets the default WinHTTP proxy configuration in the registry.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384113(v=vs.85).aspx

SetDefaultProxyConfiguration changes the proxy configuration set by ProxyCfg.exe.
The default proxy configuration set by this function can be overridden for an existing WinHTTP session by calling SetOption and specifying the OptionProxy flag. The default proxy configuration can be overridden for a new session by specifying the configuration with the WinHttpOpen function.
The dwAccessType member of the WinHTTPClientProxyInfoMBS structure pointed to by pProxyInfo should be set to kAccessTypeNamedProxy if a proxy is specified. Otherwise, it should be set to kAccessTypeDefaultProxy.
Any new sessions created after calling this function use the new default proxy configuration.
Even when WinHTTP is used in asynchronous mode (that is, when WINHTTP_FLAG_ASYNC has been set in WinHttpOpen), this function operates synchronously. The return value indicates success or failure. To get extended error information, call GetLastError.
Note For Windows XP and Windows 2000, see the Run-Time Requirements section of the WinHTTP start page.

21.14.14 Properties

21.14.15 Handle as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference.
Notes: This is a HINTERNET type.
(Read and Write property)

21.14.16 Lasterror as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error.
Notes: see also error list:
http://msdn.microsoft.com/en-us/library/aa383770(VS.85).aspx
(Read and Write property)

21.14.17 LasterrorString as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: (Read and Write property)

21.14.18 OptionConnectTimeOut as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set the connect timeout.
Notes: Lasterror is set.

Sets or retrieves an unsigned long integer value that contains the time-out value, in milliseconds. Setting this option to infinite (-1) will disable this timer.
If a TCP connection request takes longer than this time-out value, the request is canceled. The default timeout is 60 seconds. When you are attempting to connect to multiple IP addresses for a single host (a multihomed host), the timeout limit is for each individual connection.
(Read and Write property)

21.14.19 OptionProxyPassword as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set the proxy password.
Notes: Sets or retrieves a string value that contains the password used to access the proxy.
Lasterror is set.
(Read and Write property)

21.14.20 OptionProxyUsername as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Get or set the proxy user name.
Notes: Sets or retrieves a string value that contains the user name used to access the proxy.
Lasterror is set.
(Read and Write property)

21.14.21 Constants

Proxy Access Types

Constant	Value	Description
kAccessTypeDefaultProxy	0	Applies only when setting proxy information.
kAccessTypeNamedProxy	3	Internet accessed using a proxy.
kAccessTypeNoProxy	1	Internet accessed through a direct connection.

Autodetect Type Flags

Constant	Value	Description
kAutoDetectTypeDHCP	1	Use DHCP to locate the proxy auto-configuration file.
kAutoDetectTypeDNSA	2	Use DNS to attempt to locate the proxy auto-configuration file at a well-known location on the domain of the local computer.

Scheme Types

Constant	Value	Description
kInternetSchemeHTTP	1	HTTP scheme.
kInternetSchemeHTTPS	2	HTTPS scheme.

21.15   class WinHTTPClientProxyInfoMBS

21.15.1   class WinHTTPClientProxyInfoMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for the session or default proxy configuration.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa383912(v=vs.85).aspx
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.15.2   Methods

21.15.3   Constructor

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

21.15.4   Properties

21.15.5   AccessType as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The access type.
Notes: Can be kAccessTypeDefaultProxy, kAccessTypeNamedProxy or kAccessTypeNoProxy.
(Read and Write property)

21.15.6   Proxy as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The proxy server list.
Notes: (Read and Write property)

21.15.7   ProxyBypass as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The proxy bypass list.
Notes: (Read and Write property)

21.15.8   Constants

Proxy Access Types

Constant	Value	Description
kAccessTypeDefaultProxy	0	Applies only when setting proxy information.
kAccessTypeNamedProxy	3	Internet accessed using a proxy.
kAccessTypeNoProxy	1	Internet accessed through a direct connection.

21.16   class WinHTTPClientURLComponentsMBS

21.16.1   class WinHTTPClientURLComponentsMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class containing the constituent parts of an URL.
Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/aa384078(v=vs.85).aspx
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

21.16.2   Methods

21.16.3   Constructor

Plugin Version: 14.1, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

21.16.4   Properties

21.16.5   ExtraInfo as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The extra information, for example, ?something or #something.
Notes: (Read and Write property)

21.16.6   ExtraInfoLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The length of the extra information, in characters.
Notes: (Read and Write property)

21.16.7   HostName as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The host name.
Notes: (Read and Write property)

21.16.8   HostNameLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of the host name, in characters.
Notes: (Read and Write property)

21.16.9   Password as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The password.
Notes: (Read and Write property)

21.16.10   PasswordLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of the password, in characters.
Notes: (Read and Write property)

21.16.11   Port as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Port number.
Notes: (Read and Write property)

21.16.12   Scheme as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The scheme name.
Notes: (Read and Write property)

21.16.13   SchemeID as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Internet protocol scheme.
Notes: Normally 1 for HTTP and 2 for HTTPS.
(Read and Write property)

21.16.14   SchemeLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of the scheme name, in characters.
Notes: (Read and Write property)

21.16.15   UrlPath as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The URL path.
Notes: (Read and Write property)

21.16.16   UrlPathLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of the URL path, in characters.
Notes: (Read and Write property)

21.16.17   UserName as String

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The user name.
Notes: (Read and Write property)

21.16.18   UserNameLength as Integer

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of the user name, in characters.
Notes: (Read and Write property)

Chapter 22

Notifications

22.1   class WinNotificationMBS

22.1.1   class WinNotificationMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This is a class to send broadcast message to all windows applications listening for this notification.
Notes: If you have several applications, you can send notifications from one to another.

Windows Desktop application: Sending and receiving notification works.
Windows Console application: Sending works.
Windows Web application: Sending works. Receives notification of own application, but of others.

This class is useful to tell other instance of your application or other application about something. Like for example whether background process is done.

On Mac OS X you can use NSNotificationMBS class for similar functionality.

With 14.2 plugins we added possibility to listen for any windows broadcast message. Like for example for WM_TIMECHANGE.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.1
- MBS Xojo Plugins, version 17.1pr2
- MDI Window Background in Xojo
- MBS Xojo / Real Studio Plugins, version 16.2pr1

- Tip of the day: Windows Notifications

- MBS Xojo / Real Studio Plugins, version 14.2pr9

- MBS Xojo / Real Studio Plugins, version 14.2pr5

- MonkeyBread Software releases MBS Real Studio plug-ins in version 12.1

- MBS Real Studio Plugins, version 12.1pr5

- Windows Notifications

Xojo Developer Magazine

- 15.3, page 11: News

22.1.2   Methods

22.1.3   Constructor

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The default constructor.
Notes: Creates an invisible dummy window to listen for notifications.
See also:

22.1.4   Constructor(Control as DesktopUIControl)

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

Function: The control constructor.
Notes: Takes an existing Xojo control and intercepts messages going there to listen for the notifications you are interested for.
The class keeps a reference to the control to make sure it is not destroyed before this object is destroyed.
See also:

22.1.5 Constructor(control as RectControl)

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: The control constructor.
Notes: Takes an existing Xojo control and intercepts messages going there to listen for the notifications you are interested for.
The class keeps a reference to the control to make sure it is not destroyed before this object is destroyed.
See also:

22.1.6 Constructor(Window as DesktopWindow)

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

Function: The window constructor.
Notes: Takes an existing Xojo Window and intercepts messages going there to listen for the notifications you are interested for.
If you listen for global notifications sent to all windows, we prefer the other constructor.
The class keeps a reference to the window to make sure it is not destroyed before this object is destroyed.
See also:

22.1.7 Constructor(Window as window)

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: The window constructor.
Notes: Takes an existing Xojo Window and intercepts messages going there to listen for the notifications you are interested for.
If you listen for global notifications sent to all windows, we prefer the other constructor.
The class keeps a reference to the window to make sure it is not destroyed before this object is destroyed.
See also:

22.1.8 Constructor(WindowHandle as Integer)

Plugin Version: 17.1, Platform: Windows, Targets: Desktop only.

Function: The window constructor.
Notes: Takes an existing window and intercepts messages going there to listen for the notifications you are interested for.
If you listen for global notifications sent to all windows, we prefer the other constructor.
The class keeps a reference to the window to make sure it is not destroyed before this object is destroyed.
See also:

22.1.9 IsListeningFor(MessageID as Integer) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this object is listening for the given message ID.
Example:

dim w as new WinNotificationMBS

const WM_TIMECHANGE = &h001E
if w.ListenForMessage(WM_TIMECHANGE) then
MsgBox str(w.IsListeningFor(WM_TIMECHANGE))
end if

Notes: Returns true if we listen for this notification name or false if not.
See also:

22.1.10 IsListeningFor(name as string) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this object is listening for the given notification.
Example:

dim w as new WinNotificationMBS

if w.ListenForMessage("Hello") then
MsgBox str(w.IsListeningFor("Hello"))
end if

Notes: Returns true if we listen for this notification name or false if not.
See also:

22.1.11 ListenForMessage(MessageID as Integer) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Listens for given notification message id.
Notes: Returns true on success and false on failure.
See also:

22.1.12 ListenForMessage(name as string) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Listens for given notification name.
Notes: Returns true on success and false on failure.
See also:

22.1.13   SendMessage(byref result as Integer, MessageID as Integer, Value1 as Integer = 0, Value2 as Integer = 0, TimeOut as Integer = 10) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends a message to all applications listening for it.
Example:

```
dim w as new WinNotificationMBS
dim Result as Integer

// send a message and wait for result
if w.SendMessage(Result, 1234567, 123, 456) then
// okay
else
// failed
end if
```

Notes: Sends a message with given message ID.
You can pass two integer parameters.
Returns true if event was sent or false if it failed.
This doesn't tell you whether it was received by someone, but your own application receives it, too.
Timeout is in milliseconds and counts per receiving window.
Result returns the result code from the message.
See also:

22.1.14   SendMessage(name as string, Value1 as Integer = 0, Value2 as Integer = 0, TimeOut as Integer = 10) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends a message to all applications listening for it.
Example:

```
dim w as new WinNotificationMBS
```

if w.SendMessage(Result, "Hello", 123, 456) then
// okay
else
// failed
end if

Notes: You can pass two integer parameters.
Returns true if event was sent or false if it failed.
This doesn't tell you whether it was received by someone, but your own application receives it, too.
Timeout is in milliseconds and counts per receiving window.
See also:

- 22.1.13 SendMessage(byref result as Integer, MessageID as Integer, Value1 as Integer = 0, Value2 as Integer = 0, TimeOut as Integer = 10) as boolean 758

22.1.15 SendMessageToWindow(WindowHandle as Integer, byref result as Integer, MessageID as Integer, Value1 as Integer = 0, Value2 as Integer = 0, TimeOut as Integer = 10) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends a message to all applications listening for it.
Notes: Like SendMessage, but with a target window.
You can pass two integer parameters.
Returns true if event was sent or false if it failed.
This doesn't tell you whether it was received by someone, but your own application receives it, too.
Timeout is in milliseconds and counts per receiving window.

22.1.16 StopListeningForMessage(MessageID as Integer) as boolean

Plugin Version: 14.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Stops listening for a given message ID.
Example:

dim w as new WinNotificationMBS

const WM_TIMECHANGE = &h001E
if w.ListenForMessage(WM_TIMECHANGE) then
if w.StopListeningForMessage(WM_TIMECHANGE) then
MsgBox "OK"
end if
end if

See also:

22.1.17   StopListeningForMessage(name as string) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Stops listening for a given notification name.
Example:

dim w as new WinNotificationMBS

if w.ListenForMessage("Hello") then
if w.StopListeningForMessage("Hello") then
MsgBox "OK"
end if
end if

See also:

22.1.18   Properties

22.1.19   WindowHandle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal window handle.
Notes: (Read only property)

22.1.20   Events

22.1.21   GotNotification(Message as Integer, Name as string, Value1 as Integer, Value2 as Integer, byref Result as Integer, byref Handled as boolean)

Plugin Version: 12.1, Platform: Windows, Targets: .

Function: A notification was received.
Notes: Message: The message ID.

Name: If you registered by name, this is the name for the message ID.
Value1 and Value2 are parameters provided with message. (WPARAM and LPARAM)

If Handled is set to true, we pass back Result to the Windows system (LRESULT).
Else the default handler is called.

Chapter 23

Power

23.1 class WindowsPowerStateMBS

23.1.1 class WindowsPowerStateMBS

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to watch for power changes on a computer using Microsoft Windows as operation system.
Notes: Not all Windows versions call the same/all events.
Your application needs to run the event loop, so events get delivered.

This function listens for WM_POWERBROADCAST sent to all applications.

see
https://msdn.microsoft.com/en-us/library/windows/desktop/aa373247(v=vs.85).aspx

You need to run ListenForSuspendResumeNotification to receive suspend/resume events.
Blog Entries

- News from the MBS Xojo Plugins Version 23.4

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4

- MBS Xojo Plugins, version 23.4pr5

- MBS Xojo Plugins, version 23.4pr3

23.1.2   Methods

23.1.3   ListenForBatteryCapacityChanged

Plugin Version: 23.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers for battery capacity change events.
Notes: Triggers BatteryCapacityChanged event later.

23.1.4   ListenForLidSwitchStateChanged

Plugin Version: 23.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers for lid switch state change events.
Notes: Triggers LidSwitchStateChanged event later.

23.1.5   ListenForPowerSourceChanged

Plugin Version: 23.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers for power source events.
Notes: Triggers PowerSourceChanged event later.

23.1.6   ListenForSuspendResumeNotification

Plugin Version: 23.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers for suspend and resume events.

23.1.7   Events

23.1.8   BatteryCapacityChanged(Percentage as Integer)

Plugin Version: 23.4, Platform: Windows, Targets: .

Function: The remaining battery capacity has changed.
Notes: The granularity varies from system to system but the finest granularity is 1 percent. The Percentage parameter that indicates the current battery capacity remaining as a percentage from 0 through 100.

23.1.9 BatteryLow

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The BatteryLow event is broadcast to notify applications that battery power is low.
Notes: Support for this event was removed in Windows Vista bei Microsoft.

23.1.10 LidSwitchStateChanged(LidState as Integer)

Plugin Version: 23.4, Platform: Windows, Targets: .

Function: The state of the lid has changed (open or closed).
Notes: The event won't be called until a lid device is found and its current state is known.

0 - The lid is closed.
1 - The lid is opened.

23.1.11 OEMEvent(eventcode as Integer)

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The OEMEvent event is broadcast when an APM BIOS signals an APM OEM event.
Notes: eventcode: A DWORD value that specifies the OEM-defined event code that was signaled by the system's APM BIOS. OEM event codes are in the range &h0200 - &h02FF.

Because not all APM BIOS implementations provide OEM event notifications, this event may never be broadcast on some computers.

Support for this event was removed in Windows Vista bei Microsoft.

23.1.12 PowerSettingChange(data as MemoryBlock)

Plugin Version: 23.4, Platform: Windows, Targets: .

Function: A power setting change event has been received.
Notes: This is the raw event with data.
We also call BatteryCapacityChanged, PowerSourceChanged and LidSwitchStateChanged events if their GUIDs are happening.

The data memoryblock contains POWERBROADCAST_SETTING data, where first 16 bytes are the GUID for the type of change. Then follows the 32-bit integer for data length and the data specific to the event.

see
https://learn.microsoft.com/en-us/windows/win32/api/winuser/ns-winuser-powerbroadcast_setting

23.1.13   PowerSourceChanged(power as Integer)

Plugin Version: 23.4, Platform: Windows, Targets: .

Function: The system power source has changed.
Notes: PowerSourceAC (0) - The computer is powered by an AC power source (or similar, such as a laptop powered by a 12V automotive adapter).
PowerSourceDC (1) - The computer is powered by an onboard battery power source.
PowerSourceUPS (2) - The computer is powered by a short-term power source such as a UPS device.

Only called if you called ListenForPowerSourceChanged once.

23.1.14   PowerStatusChange

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The PowerStatusChange event is broadcast when a change in the power status of the computer is detected, such as a switch from battery power to A/C.
Notes: The system also broadcasts this event when remaining battery power slips below the threshold specified by the user or if the battery power changes by a specified percentage.

23.1.15   QueryStandby(PromptUser as boolean) as boolean

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The QueryStandby event is broadcast to request permission to set the computer into standby mode.
Notes: An application that grants permission should carry out preparations for the suspension before returning.

PromptUser: If true, the application can prompt the user for directions on how to prepare for the suspension; otherwise, the application must prepare without user interaction.

Return true to deny the request, false to grant the request.

An application should process this event by first determining whether permission to go to standby mode can

be granted. It must not grant permission if doing so would cause a loss of data. The application can prompt the user for directions on how to prepare for suspension only if PromptUser is set.

The system allows approximately 20 seconds for an application to remove the message that is sending the QueryStandby event from the application's message queue. If an application does not remove the message from its queue in less then 20 seconds, the system will assume that the application is in a non-responsive state, and that the application agrees to the sleep request. Applications that do not process their message queues may have their operations interrupted. After it removes the message from the message queue, an application can take as much time as needed to perform any required operations before entering the sleep state. Any operations that could take longer then 20 seconds should be performed at this time, since the system allows only 20 seconds for operations to complete during standby processing.

23.1.16   QueryStandbyFailed

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The QueryStandbyFailed event is broadcast as a notification that permission to go to standby mode was denied.
Notes: This event is broadcast if any application or driver returned false to a previous QueryStandby event.

23.1.17   QuerySuspend(PromptUser as boolean) as boolean

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The QuerySuspend event is broadcast to request permission to suspend the computer.
Notes: An application that grants permission should carry out preparations for the suspension before returning.

PromptUser: If true, the application can prompt the user for directions on how to prepare for the suspension; otherwise, the application must prepare without user interaction.

Return true to deny the request, false to grant the request.

An application should process this event by first determining whether permission to suspend can be granted. It must not grant permission if doing so would cause a loss of data. The application can prompt the user for directions on how to prepare for suspension only if PromptUser is set.

The system allows approximately 20 seconds for an application to remove the message that is sending the QuerySuspend event from the application's message queue. If an application does not remove the message from its queue in less then 20 seconds, the system will assume that the application is in a non-responsive state, and that the application agrees to the sleep request. Applications that do not process their message queues may have their operations interrupted. After it removes the message from the message queue, an

application can take as much time as needed to perform any required operations before entering the sleep state. Any operations that could take longer then 20 seconds should be performed at this time, since the system allows only 20 seconds for operations to complete during Suspend processing.

Support for this event was removed in Windows Vista bei Microsoft.

23.1.18   QuerySuspendFailed

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The QuerySuspendFailed event is broadcast as a notification that permission to suspend the computer was denied.
Notes: This event is broadcast if any application or driver returned false to a previous QuerySuspend event.

Support for this event was removed in Windows Vista bei Microsoft.

23.1.19   ResumeAutomatic

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The ResumeAutomatic event is broadcast when the computer wakes up automatically to handle an event.
Notes: An application will not generally respond unless it is handling the event, because the user is not present.

23.1.20   ResumeCritical

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The ResumeCritical event is broadcast as a notification that the system has resumed operation.
Notes: This event can indicate that some or all applications did not receive a Suspend event. For example, this event can be broadcast after a critical suspension caused by a failing battery.

Support for this event was removed in Windows Vista bei Microsoft.

23.1.21   ResumeStandby

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The ResumeStandby event is broadcast as a notification that the system has resumed operation after being put to standby.

23.1.22   ResumeSuspend

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The ResumeSuspend event is broadcast as a notification that the system has resumed operation after being suspended.

23.1.23   Standby

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The Standby event is broadcast immediately before the computer enters a standby state.
Notes: This event is typically broadcast when all applications and installable drivers have returned okay to a previous QueryStandby event.

23.1.24   Suspend

Plugin Version: 6.1, Platform: Windows, Targets: .

Function: The Suspend event is broadcast immediately before the computer enters a suspended state.
Notes: This event is typically broadcast when all applications and installable drivers have returned okay to a previous QuerySuspend event.

23.1.25   Constants

Lid States

Constant	Value	Description
LidClosed	0	The lid is closed.
LidOpened	1	The lid is opened.

Power Source States

Constant	Value	Description
PowerSourceAC	0	The computer is powered by an AC power source (or similar, such as a laptop powered by a 12V automotive adapter).
PowerSourceDC	1	The computer is powered by an onboard battery power source.
PowerSourceUPS	2	The computer is powered by a short-term power source such as a UPS device.

Chapter 24

Printing

24.1   class WindowsAddPrintJobMBS

24.1.1   class WindowsAddPrintJobMBS

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to send raw data to a printer.
Example:

```
// Print Postscript directly to Postscript printer
const PrinterName = "Brother DCP-8085DN"

dim w as new WindowsAddPrintJobMBS

if not w.OpenPrinter(PrinterName) then
MsgBox "OpenPrinter failed. Is the printer name correct in the source code?"
Return
end if

const DocName = "My Document"

if not w.StartDocPrinter("My Document", w.kDataFormatRAW) then
MsgBox "StartDocPrinter failed."
Return
end if

MsgBox "Print Job ID: "+str(w.JobID)

call w.StartPagePrinter

dim PostScript as string = "%!PS"+EndOfLine.UNIX+".1 setgray"+EndOfLine.UNIX+"0 0 100 100 rect-
```

fill"+EndOfLine.UNIX+"showpage"+EndOfLine.UNIX

dim BytesSent as Integer = w.WritePrinter(PostScript)

MsgBox str(BytesSent)+" bytes of "+str(lenb(PostScript))+" bytes sent."

call w.EndPagePrinter
call w.EndDocPrinter

w = nil // close printer

Notes: Perfect for printing postscript data to a postscript printer.
You can use this class in several ways:

1. OpenPrinter, AddJob, WriteJob and ScheduleJob.
2. OpenPrinter, StartDocPrinter, StartPagePrinter, WritePrinter, EndPagePrinter, EndDocPrinter and ClosePrinter.
3. OpenPrinter, StartDocPrinter, WritePrinter, EndDocPrinter and ClosePrinter.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.2pr1

- More on printing on Windows

- MBS Plugins 10.3 Release Notes

24.1.2   Methods

24.1.3   AddJob as boolean

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new print job.
Notes: Returns true on success and false on failure.

24.1.4   ClosePrinter

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Closes printer connection.

24.1.5 EndDocPrinter as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The EndDocPrinter function ends a print job for the specified printer.
Notes: If the function succeeds, the return value is a true.

The EndDocPrinter function returns an error if the print job was not started by calling the StartDocPrinter function.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it may fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.

24.1.6 EndPagePrinter as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The EndPagePrinter function notifies the print spooler that the application is at the end of a page in a print job.
Notes: If the function succeeds, the return value is a true.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error

message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.

24.1.7   OpenPrinter(PrinterName as string) as boolean

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens the printer for printing.
Notes: Returns true on success and false on failure.
PrinterName can be "" for the default printer.

24.1.8   ScheduleJob as boolean

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Schedules the printing.
Notes: Returns true on success and false on failure.

24.1.9   StartDocPrinter(DocName as string, Datatype as string) as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StartDocPrinter function notifies the print spooler that a document is to be spooled for printing.
Example:

```
// Print Postscript directly to Postscript printer
const PrinterName = "Brother DCP-8085DN"

dim w as new WindowsAddPrintJobMBS

if not w.OpenPrinter(PrinterName) then
MsgBox "OpenPrinter failed. Is the printer name correct in the source code?"
Return
end if

const DocName = "My Document"

if not w.StartDocPrinter("My Document", w.kDataFormatRAW) then
MsgBox "StartDocPrinter failed."
Return
end if
```

MsgBox "Print Job ID: "+str(w.JobID)

call w.StartPagePrinter

dim PostScript as string = "%!PS"+EndOfLine.UNIX+".1 setgray"+EndOfLine.UNIX+"0 0 100 100 rectfill"+EndOfLine.UNIX+"showpage"+EndOfLine.UNIX

dim BytesSent as Integer = w.WritePrinter(PostScript)

MsgBox str(BytesSent)+" bytes of "+str(lenb(PostScript))+" bytes sent."

call w.EndPagePrinter
call w.EndDocPrinter

w = nil // close printer

Notes: DocName: A string that specifies the name of the document.
OutputFile: Optional a path string or a folderitem that specifies the name of an output file. To print to a printer, set this to "" or nil or leave away.
Datatype: A string that identifies the type of data used to record the document.

If the function succeeds, the return value is true and the JobID property is set.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.
See also:

- 24.1.10 StartDocPrinter(DocName as string, OutputFile as folderitem, Datatype as string) as boolean 776

- 24.1.11 StartDocPrinter(DocName as string, OutputFilePath as string, Datatype as string) as boolean 777

24.1.10   StartDocPrinter(DocName as string, OutputFile as folderitem, Datatype as string) as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StartDocPrinter function notifies the print spooler that a document is to be spooled for printing.
Example:

```
// Print Postscript directly to Postscript printer
const PrinterName = "Brother DCP-8085DN"

dim w as new WindowsAddPrintJobMBS

if not w.OpenPrinter(PrinterName) then
MsgBox "OpenPrinter failed. Is the printer name correct in the source code?"
Return
end if

const DocName = "My Document"

// output to a file
dim file as FolderItem = SpecialFolder.Desktop.Child("test.ps")

if not w.StartDocPrinter("My Document", file, w.kDataFormatRAW) then
MsgBox "StartDocPrinter failed."
Return
end if

MsgBox "Print Job ID: "+str(w.JobID)

call w.StartPagePrinter

dim PostScript as string = "%!PS"+EndOfLine.UNIX+".1 setgray"+EndOfLine.UNIX+"0 0 100 100 rect-
fill"+EndOfLine.UNIX+"showpage"+EndOfLine.UNIX

dim BytesSent as Integer = w.WritePrinter(PostScript)

MsgBox str(BytesSent)+" bytes of "+str(lenb(PostScript))+" bytes sent."

call w.EndPagePrinter
call w.EndDocPrinter

w = nil // close printer
```

Notes: DocName: A string that specifies the name of the document.
OutputFile: Optional a path string or a folderitem that specifies the name of an output file. To print to a

printer, set this to "" or nil or leave away.
Datatype: A string that identifies the type of data used to record the document.

If the function succeeds, the return value is true and the JobID property is set.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.
See also:

- 24.1.9 StartDocPrinter(DocName as string, Datatype as string) as boolean   774

- 24.1.11 StartDocPrinter(DocName as string, OutputFilePath as string, Datatype as string) as boolean 777

24.1.11   StartDocPrinter(DocName as string, OutputFilePath as string, Datatype as string) as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StartDocPrinter function notifies the print spooler that a document is to be spooled for printing.
Example:

```
// Print Postscript directly to Postscript printer
const PrinterName = "Brother DCP-8085DN"

dim w as new WindowsAddPrintJobMBS

if not w.OpenPrinter(PrinterName) then
MsgBox "OpenPrinter failed. Is the printer name correct in the source code?"
Return
end if

const DocName = "My Document"
```

```
// output to a file
dim file as string = "C:\test.ps"

if not w.StartDocPrinter("My Document", file, w.kDataFormatRAW) then
MsgBox "StartDocPrinter failed."
Return
end if

MsgBox "Print Job ID: "+str(w.JobID)

call w.StartPagePrinter

dim PostScript as string = "%!PS"+EndOfLine.UNIX+".1 setgray"+EndOfLine.UNIX+"0 0 100 100 rect-
fill"+EndOfLine.UNIX+"showpage"+EndOfLine.UNIX

dim BytesSent as Integer = w.WritePrinter(PostScript)

MsgBox str(BytesSent)+" bytes of "+str(lenb(PostScript))+" bytes sent."

call w.EndPagePrinter
call w.EndDocPrinter

w = nil // close printer
```

Notes: DocName: A string that specifies the name of the document.
OutputFile: Optional a path string or a folderitem that specifies the name of an output file. To print to a printer, set this to "" or nil or leave away.
Datatype: A string that identifies the type of data used to record the document.

If the function succeeds, the return value is true and the JobID property is set.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling

processes, and the amount of fragmentation in the process heap.
See also:

24.1.12   StartPagePrinter as boolean

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The StartPagePrinter function notifies the spooler that a page is about to be printed on the specified printer.
Notes: If the function succeeds, the return value is a true.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.

24.1.13   WriteJob(data as string) as Integer

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds data to the spool file.
Notes: You need to pass raw data for the printer, e.g. postscript data.
Returns the number of bytes written.
So lenb(data) should be equal to the result.
Only available between a call to AddJob and ScheduleJob.

24.1.14   WritePrinter(data as string) as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WritePrinter function notifies the print spooler that data should be written to the specified printer.
Notes: data: The data that should be written to the printer.

If the function succeeds, the return value is a true.

The sequence for a print job is as follows:

1. To begin a print job, call StartDocPrinter.
2. To begin each page, call StartPagePrinter.
3. To write data to a page, call WritePrinter.
4. To end each page, call EndPagePrinter.
5. Repeat 2, 3, and 4 for as many pages as necessary.
6. To end the print job, call EndDocPrinter.

When a high-level document (such as an Adobe PDF or Microsoft Word file) or other printer data (such PCL, PS, or HPGL) is sent directly to a printer, the print settings defined in the document take precedent over Windows print settings.

When a page in a spooled file exceeds approximately 350 MB, it can fail to print and not send an error message. For example, this can occur when printing large EMF files. The page size limit depends on many factors including the amount of virtual memory available, the amount of memory allocated by calling processes, and the amount of fragmentation in the process heap.

24.1.15   Properties

24.1.16   JobID as Integer

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The JobID of this print.
Notes: For information only.
Maybe be useful for the user to find this print job later.
(Read and Write property)

24.1.17 JobPath as String

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file name of the temporary file used.
Notes: Just for information and only available between a call to AddJob and ScheduleJob.
(Read and Write property)

24.1.18 lastError as Integer

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Zero for no error.
-1 if the plugin has a problem like invalid parameter or called on Mac/Linux.
others are Windows error codes.
(Read and Write property)

24.1.19 lastErrorMessage as String

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The localized text message for LastError property.
Notes: (Read and Write property)

24.1.20 PrinterHandle as Integer

Plugin Version: 6.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the printer.
Notes: (Read and Write property)

24.1.21 Constants

Constants

Constant	Value	Description
kDataFormatRAW	"RAW"	One of the data format constants.
		Use this constant if you send to both local and networked printers that us GDI-based printer drivers.
kDataFormatXPS_PASS	"XPS_PASS"	One of the data format constants.
		Use this constant to send printer control data directly to printers that us XPSDrv printer drivers.

24.2 class WindowsDeviceModeMBS

24.2.1 class WindowsDeviceModeMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for device modue information for a Windows printer.
Example:

```
dim d as new WindowsDeviceModeMBS
dim name as string = "My printer"

// here we define which values we want to change
d.Fields = d.DM_ORIENTATION
// and change value
d.Orientation = d.DMORIENT_PORTRAIT

dim w as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name)
if w.ChangePrinterSettings(d, 2) then
MsgBox "OK"
else
MsgBox "Failed"
end if
```

Notes: Basicly this class wraps the Windows DEVMODE structure.

For more information on the DEVMODE structure, visit this website:
http://msdn.microsoft.com/en-us/library/dd183565(VS.85).aspx
Blog Entries

- MBS Xojo Plugins, version 17.6pr4

- Detect PrinterSetup data format

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.3

- MBS Xojo Plugins, version 17.3pr7

- MBS Xojo Plugins, version 17.2pr3

- Print to Printer in Xojo on Windows

- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.3

- Updating PrinterSetup.SetupString on Windows

- More on printing on Windows

- MBS REALbasic Plugins Version 10.4 release notes

Videos

- Presentation from Munich conference about MBS Plugins.

Xojo Developer Magazine

- 15.5, page 9: News

24.2.2   Methods

24.2.3   ApplyToSetupString(SetupString as String) as string

Plugin Version: 17.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Applies current device mode to the given setup string.
Example:

dim p as new PrinterSetup

if p.PageSetupDialog then

dim w as WindowsDeviceModeMBS = WindowsDeviceModeMBS.FromSetupString(s)

w.Copies = 3
w.Fields = BitwiseOr(w.Fields, w.DM_COPIES)

dim s as string = w.SetupString
dim t as string = w.ApplyToSetupString(s)

Break // check

end if

Notes: The setup string contains a device mode and we replace the one from Xojo with our own.
The other information is kept the same.

24.2.4   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new empty device mode.
Notes: Size is set for you to default name of the data structure.

24.2.5 FromRawData(data as memoryblock, Unicode as boolean = true) as WindowsDeviceModeMBS

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new WindowsDeviceModeMBS object with given data.
Notes: The data is expected to the in format of DEVMODEW structure with some extra data and the right values inside.
If Unicode is true, we use DEVMODEW and if Unicode is false, we use DEVMODEA structure.
See also:

- 24.2.6 FromRawData(data as string, Unicode as boolean = true) as WindowsDeviceModeMBS   785

24.2.6 FromRawData(data as string, Unicode as boolean = true) as Windows-DeviceModeMBS

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new WindowsDeviceModeMBS object with given data.
Notes: The data is expected to the in format of DEVMODEW structure with some extra data and the right values inside.
If Unicode is true, we use DEVMODEW and if Unicode is false, we use DEVMODEA structure.
See also:

- 24.2.5 FromRawData(data as memoryblock, Unicode as boolean = true) as WindowsDeviceModeMBS 785

24.2.7 FromSetupString(SetupString as String) as WindowsDeviceModeMBS

Plugin Version: 14.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initialized DevMode object from setup string.
Example:

dim p as new PrinterSetup

// setup something
call p.PageSetupDialog

// now we have nice setupstring
dim ss as string = p.SetupString

// parse it in device mode
dim d as WindowsDeviceModeMBS = WindowsDeviceModeMBS.FromSetupString(ss)

// duplex is?
MsgBox "Duplex: "+str(d.Duplex)

```
// change printer
d.DeviceName = "Deskjet 2540 series#:2"

// enable duplex
d.Fields = BitwiseOr(d.Fields, d.DM_DUPLEX)
d.Duplex = d.DMDUP_HORIZONTAL

// now duplex is?
MsgBox "Duplex: "+str(d.Duplex)

// get back as setup string
dim da as string = d.SetupString

if da = "" then
MsgBox "failed to create setup string"
Return
end if

// assign back
p.SetupString = da

// take a look
call p.PageSetupDialog

// and print something
dim g as Graphics = OpenPrinter(p)

g.DrawString "Page 1", 20, 20
g.NextPage
g.DrawString "Page 2", 20, 20
```

Notes: You can pass string from PrinterSetup.SetupString.

Supports Xojo 2017r1 with 17.3 plugins.

24.2.8   RawData(Unicode as boolean = true) as memoryblock

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies the raw data of the WindowsDeviceModeMBS class.
Notes: Returns nil on any error.
If Unicode is true, we use DEVMODEW and if Unicode is false, we use DEVMODEA structure.

24.2.9 SetupString(ActualHorizontalResolution as integer, ActualVerticalResolution as integer, MaxHorizontalResolution as integer, MaxVerticalResolution as integer, MarginLeft as integer = 2500, MarginRight as integer = 2500, MarginTop as integer = 2500, MarginBottom as integer = 2500, MinMarginLeft as integer = 0, MinMarginRight as integer = 0, MinMarginTop as integer = 0, MinMarginBottom as integer = 0, PageSetupFlags as integer = 8) as string

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a setup string from devmode.
Notes: This setupstring is empty on error.
If not empty, you can assign to PrinterSetup.SetupString.

Please pass reasonable flags.
Check existing SetupStrings from Xojo for possible values.
Resolution parameters can e.g. bei 72.

Supports Xojo 2017r1 with 17.3 plugins.
See also:

24.2.10 SetupString(Margin as Integer = 2500) as string

Plugin Version: 14.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a setup string from devmode.
Example:

dim p as new PrinterSetup

// setup something
call p.PageSetupDialog

// now we have nice setupstring
dim ss as string = p.SetupString

// parse it in device mode
dim d as WindowsDeviceModeMBS = WindowsDeviceModeMBS.FromSetupString(ss)

// duplex is?
MsgBox "Duplex: "+str(d.Duplex)

// change printer
d.DeviceName = "Deskjet 2540 series#:2"

```
// enable duplex
d.Fields = BitwiseOr(d.Fields, d.DM_DUPLEX)
d.Duplex = d.DMDUP_HORIZONTAL

// now duplex is?
MsgBox "Duplex: "+str(d.Duplex)

// get back as setup string
dim da as string = d.SetupString

if da = "" then
MsgBox "failed to create setup string"
Return
end if

// assign back
p.SetupString = da

// take a look
call p.PageSetupDialog

// and print something
dim g as Graphics = OpenPrinter(p)

g.DrawString "Page 1", 20, 20
g.NextPage
g.DrawString "Page 2", 20, 20
```

Notes: This setupstring is empty on error.
If not empty, you can assign to PrinterSetup.SetupString.

Supports Xojo 2017r1 with 17.3 plugins.
See also:

24.2.11 Properties

24.2.12 Collate as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies whether collation should be used when printing multiple copies.
Notes: This member is ignored unless the printer driver indicates support for collation by setting the dm-Fields member to DM_COLLATE.
This member can be one of the following values:

Value	Meaning
DMCOLLATE_TRUE	Collate when printing multiple copies.
DMCOLLATE_FALSE	Do not collate when printing multiple copies.

(Read and Write property)

24.2.13 Color as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Switches between color and monochrome on color printers.
Notes: The following are the possible values:
DMCOLOR_COLOR
DMCOLOR_MONOCHROME
(Read and Write property)

24.2.14 Copies as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Selects the number of copies printed if the device supports multiple-page copies.
Example:

dim w as WindowsDeviceModeMBS // your device mode object

w.Copies = 3
w.Fields = BitwiseOr(w.Fields, w.DM_COPIES)

Notes: (Read and Write property)

24.2.15   Data as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The raw data pointer to the DEVMODEW structure.
Notes: (Read only property)

24.2.16   DefaultSource as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the paper source.
Notes: Use the DMBIN_* constants.
(Read and Write property)

24.2.17   DeviceName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the "friendly" name of the printer or display; for example, "PCL/HP LaserJet" in the case of PCL/HP LaserJet.
Notes: This string is unique among device drivers. Note that this name may be truncated to fit in the dmDeviceName field.
(Read and Write property)

24.2.18   DitherType as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies how dithering is to be done.
Notes: The member can be one of the following predefined values, or a driver-defined value greater than or equal to the value of DMDITHER_USER.

Value	Meaning
DMDITHER_NONE	No dithering.
DMDITHER_COARSE	Dithering with a coarse brush.
DMDITHER_FINE	Dithering with a fine brush.
DMDITHER_LINEART	Line art dithering, a special dithering method that produces well defined borders between black, white, and gray scaling. It is not suitable for images that include continuous graduations in intensity and hue, such as scanned photographs.
DMDITHER_GRAYSCALE	Device does gray scaling.

(Read and Write property)

24.2.19   DriverExtra as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Contains the number of bytes of private driver-data that follow this structure. If a device driver does not use device-specific information, set this member to zero.
Notes: (Read and Write property)

24.2.20   DriverVersion as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The driver version number assigned by the driver developer.
Notes: (Read and Write property)

24.2.21   Duplex as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Selects duplex or double-sided printing for printers capable of duplex printing.
Notes: Following are the possible values.

Value	Meaning
DMDUP_SIMPLEX	Normal (nonduplex) printing.
DMDUP_HORIZONTAL	Short-edge binding, that is, the long edge of the page is horizontal.
DMDUP_VERTICAL	Long-edge binding, that is, the long edge of the page is vertical.

(Read and Write property)

24.2.22   Fields as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies whether certain properties have been initialized.
Notes: If a property is initialized, its corresponding bit is set, otherwise the bit is clear. A driver supports only those properties that are appropriate for the printer or display technology.

Use the DM_* constants which match by name their property. So the DM_ORIENTATION bit value is related to the Orientation property.
(Read and Write property)

24.2.23   FormName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string array that specifies the name of the form to use; for example, "Letter" or "Legal".
Notes: A complete set of names can be retrieved by using the GetPrinterFormats function on the WindowsPrinterListMBS class.
(Read and Write property)

24.2.24   ICMIntent as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies which color matching method, or intent, should be used by default.
Notes: This member is primarily for non-ICM applications. ICM applications can establish intents by using the ICM functions. This member can be one of the following predefined values, or a driver defined value greater than or equal to the value of DMICM_USER.

Value	Meaning
DMICM_ABS_COLORIMETRIC	Color matching should optimize to match the exact color requested without white point mapping. This value is most appropriate for use with proofing.
DMICM_COLORIMETRIC	Color matching should optimize to match the exact color requested. This value is most appropriate for use with business logos or other images when an exact color match is desired.
DMICM_CONTRAST	Color matching should optimize for color contrast. This value is the most appropriate choice for scanned or photographic images when dithering is desired.
DMICM_SATURATE	Color matching should optimize for color saturation. This value is the most appropriate choice for business graphs when dithering is not desired.

(Read and Write property)

24.2.25   ICMMethod as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies how ICM is handled.
Notes: For a non-ICM application, this member determines if ICM is enabled or disabled. For ICM applications, the system examines this member to determine how to handle ICM support. This member can

be one of the following predefined values, or a driver-defined value greater than or equal to the value of DMICMMETHOD_USER.

Value	Meaning
DMICMMETHOD_NONE	Specifies that ICM is disabled.
DMICMMETHOD_SYSTEM	Specifies that ICM is handled by Windows.
DMICMMETHOD_DRIVER	Specifies that ICM is handled by the device driver.
DMICMMETHOD_DEVICE	Specifies that ICM is handled by the destination device.

The printer driver must provide a user interface for setting this member. Most printer drivers support only the DMICMMETHOD_SYSTEM or DMICMMETHOD_NONE value. Drivers for PostScript printers support all values.
(Read and Write property)

24.2.26   LogPixels as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of pixels per logical inch.
Notes: Printer drivers do not use this member.
(Read and Write property)

24.2.27   MediaType as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the type of media being printed on.
Notes: The member can be one of the following predefined values, or a driver-defined value greater than or equal to the value of DMMEDIA_USER.

Value	Meaning
DMMEDIA_STANDARD	Plain paper.
DMMEDIA_GLOSSY	Glossy paper.
DMMEDIA_TRANSPARENCY	Transparent film.

(Read and Write property)

24.2.28   Nup as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies where the NUP is done.
Notes: It can be one of the following.

Value	Meaning
DMNUP_SYSTEM	The print spooler does the NUP.
DMNUP_ONEUP	The application does the NUP.

(Read and Write property)

24.2.29   Orientation as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printer devices only, selects the orientation of the paper.
Example:

```
dim d as new WindowsDeviceModeMBS
dim name as string = "My printer"

// here we define which values we want to change
d.Fields = d.DM_ORIENTATION
// and change value
d.Orientation = d.DMORIENT_PORTRAIT

dim w as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name)
if w.ChangePrinterSettings(d, 2) then
MsgBox "OK"
else
MsgBox "Failed"
end if
```

Notes: This member can be either DMORIENT_PORTRAIT (1) or DMORIENT_LANDSCAPE (2).
(Read and Write property)

24.2.30   PaperLength as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printer devices only, overrides the length of the paper specified by the PaperSize member, either for custom paper sizes or for devices such as dot-matrix printers that can print on a page of arbitrary length.

Notes: These values, along with all other values in this structure that specify a physical length, are in tenths of a millimeter.
(Read and Write property)

24.2.31 PaperSize as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printer devices only, selects the size of the paper to print on.
Notes: This member can be set to zero if the length and width of the paper are both set by the PaperLength and PaperWidth members. Otherwise, the PaperSize member can be set to a device specific value greater than or equal to DMPAPER_USER or to one of the following predefined values with DMPAPER_* constants.
(Read and Write property)

24.2.32 PaperWidth as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printer devices only, overrides the width of the paper specified by the PaperSize member.
Notes: Unit is tenths of a millimeter.
(Read and Write property)

24.2.33 PrintQuality as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the printer resolution.
Notes: There are four predefined device-independent values:

DMRES_HIGH
DMRES_MEDIUM
DMRES_LOW
DMRES_DRAFT

If a positive value is specified, it specifies the number of dots per inch (DPI) and is therefore device dependent.
(Read and Write property)

24.2.34   Scale as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the factor by which the printed output is to be scaled.
Notes: The apparent page size is scaled from the physical page size by a factor of Scale /100. For example, a letter-sized page with a dmScale value of 50 would contain as much data as a page of 17- by 22-inches because the output text and graphics would be half their original height and width.
(Read and Write property)

24.2.35   Size as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the size, in bytes, of the DEVMODE structure, not including any private driver-specific data that might follow the structure's public members.
Notes: You don't need to set this field normally as the plugin does that for you.
(Read and Write property)

24.2.36   SpecVersion as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The version number of the initialization data specification on which the structure is based.
Notes: You don't need to set this field normally as the plugin does that for you.
(Read and Write property)

24.2.37   TTOption as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies how TrueType fonts should be printed.
Notes: This member can be one of the following values:

Value	Meaning
DMTT_BITMAP	Prints TrueType fonts as graphics. This is the default action for dot-matrix printers.
DMTT_DOWNLOAD	Downloads TrueType fonts as soft fonts. This is the default action for Hewlett-Packard printers that use Printer Control Language (PCL).
DMTT_DOWNLOAD_OUTLINE	Downloads TrueType fonts as outline soft fonts.
DMTT_SUBDEV	Substitutes device fonts for TrueType fonts. This is the default action for PostScript printers.

(Read and Write property)

24.2.38 YResolution as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the y-resolution, in dots per inch, of the printer.
Notes: If the printer initializes this member, the PrintQuality member specifies the x-resolution, in dots per inch, of the printer.
(Read and Write property)

24.2.39 Constants

Constants

Constant	Value	Description
DMBIN_AUTO	7	One of the source bin constants. Auto
DMBIN_CASSETTE	14	One of the source bin constants. Cassette
DMBIN_ENVELOPE	5	One of the source bin constants. Envelope
DMBIN_ENVMANUAL	6	One of the source bin constants. Envelope Manual.
DMBIN_FORMSOURCE	15	One of the source bin constants. Form Source.
DMBIN_LARGECAPACITY	11	One of the source bin constants. Large Capacity
DMBIN_LARGEFMT	10	One of the source bin constants. Large Format
DMBIN_LOWER	2	One of the source bin constants. Lower
DMBIN_MANUAL	4	One of the source bin constants. Manual
DMBIN_MIDDLE	3	One of the source bin constants. Middle
DMBIN_ONLYONE	1	One of the source bin constants. Only One.
DMBIN_SMALLFMT	9	One of the source bin constants. Small Format
DMBIN_TRACTOR	8	One of the source bin constants. Tractor
DMBIN_UPPER	1	One of the source bin constants. Upper
DMBIN_USER	256	One of the source bin constants. User
DMCOLLATE_FALSE	0	One of the collate mode constants. Do not collate when printing multiple copies
DMCOLLATE_TRUE	1	One of the collate mode constants. Collate when printing multiple copies.
DMCOLOR_COLOR	2	One of the color mode constants. Color
DMCOLOR_MONOCHROME	1	One of the color mode constants. Monochrome.
DMDITHER_COARSE	2	One of the dither constants. Dithering with a coarse brush.
DMDITHER_ERRORDIFFUSION	5	One of the dither constants.
DMDITHER_FINE	3	One of the dither constants. Dithering with a fine brush.
DMDITHER_GRAYSCALE	10	One of the dither constants. Device does gray scaling.
DMDITHER_LINEART	4	One of the dither constants. Line art dithering, a special dithering metho ders between black, white, and gray scalin that include continuous graduations in intens tographs.
DMDITHER_NONE	1	One of the dither constants. No dithering.
DMDITHER_RESERVED6	6	One of the dither constants.
DMDITHER_RESERVED7	7	One of the dither constants.
DMDITHER_RESERVED8	8	One of the dither constants.
DMDITHER_RESERVED9	9	One of the dither constants.
DMDITHER_USER	256	One of the dither constants.
DMDUP_HORIZONTAL	3	One of the duplex mode constants.

24.3   class WindowsGraphicsInfoMBS

24.3.1   class WindowsGraphicsInfoMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class gives information for Windows Device contexts.
Example:

dim g as Graphics = OpenPrinter
dim info as WindowsGraphicsInfoMBS = g.WindowsGraphicsInfoMBS

MsgBox _
"Printer physical page size: "+str(info.PhysicalWidth)+" "+str(info.PhysicalHeight)+_
EndOfLine+_
"Useable paper size: "+str(info.ResolutionX)+ " x "+str(info.ResolutionY)+_
EndOfLine+_
"Paper size: "+str(info.SizeX)+" x "+str(info.Sizey)+" mm"

Notes: Main use currently is to learn about native resolution of a printer graphics object.
Blog Entries

- MBS Xojo Plugins 18.3

- MBS Xojo Plugins, version 18.3pr5

- MBS Real Studio Plugins, version 12.1pr6

- More on printing on Windows

Xojo Developer Magazine

- 16.5, page 9: News

24.3.2   Methods

24.3.3   Constructor

Plugin Version: 18.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initializes properties for main screen.
See also:

24.3.4   Constructor(c as control)

Plugin Version: 18.3, Platform: Windows, Targets: Desktop only.

Function: Initializes properties for this control object.
See also:

24.3.5   Constructor(c as DesktopControl)

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

Function: Initializes properties for this control object.
See also:

24.3.6   Constructor(g as graphics)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Initializes properties for this graphics object.
See also:

24.3.7   Constructor(p as Picture)

Plugin Version: 18.3, Platform: Windows, Targets: Desktop only.

Function: Initializes properties for this picture object.
See also:

24.3.8   Constructor(w as DesktopWindow)

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

Function: Initializes properties for this window object.
See also:

24.3.9   Constructor(w as window)

Plugin Version: 18.3, Platform: Windows, Targets: Desktop only.

Function: Initializes properties for this window object.
See also:

24.3.10   Properties

24.3.11   AspectX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Relative width of a device pixel used for line drawing.
Notes: (Read and Write property)

24.3.12   AspectXY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Diagonal width of the device pixel used for line drawing.
Notes: (Read and Write property)

24.3.13   AspectY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Relative height of a device pixel used for line drawing.
Notes: (Read and Write property)

24.3.14 BitsPerPixel as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of adjacent color bits for each pixel.
Example:

dim p as new Picture(200, 200, 32)
dim info as new WindowsGraphicsInfoMBS(p.Graphics)

MsgBox "Color Depth: "+str(info.BitsPerPixel)

Notes: As example gives a display device context, the bits per pixel is the screen depth.
(Read and Write property)

24.3.15 BrushesCount as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of device-specific brushes.
Notes: (Read and Write property)

24.3.16 ColorCount as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of entries in the device's color table, if the device has a color depth of no more than 8 bits per pixel.
Notes: For devices with greater color depths, 1 is returned.
(Read and Write property)

24.3.17 DesktopResolutionX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Horizontal width of entire desktop in pixels.
Example:

dim p as new Picture(200, 200, 32)
dim info as new WindowsGraphicsInfoMBS(p.Graphics)

MsgBox "Size of Windows Desktop: "+str(info.DesktopResolutionX)+ " x "+str(info.DesktopResolutionY)

Notes: (Read and Write property)

24.3.18   DesktopResolutionY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Vertical height of entire desktop in pixels.
Example:

dim p as new Picture(200, 200, 32)
dim info as new WindowsGraphicsInfoMBS(p.Graphics)

MsgBox "Size of Windows Desktop: "+str(info.DesktopResolutionX)+ " x "+str(info.DesktopResolutionY)

Notes: (Read and Write property)

24.3.19   DriverVersion as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device driver version.
Notes: (Read and Write property)

24.3.20   FontCount as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of device-specific fonts.
Notes: (Read and Write property)

24.3.21   LogPixelsX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of pixels per logical inch along the screen width.
Example:

dim g as Graphics = OpenPrinter
dim info as WindowsGraphicsInfoMBS = g.WindowsGraphicsInfoMBS

MsgBox "Printer resolution: "+str(info.LogPixelsX)+" x "+str(info.LogPixelsY)

Notes: In a system with multiple display monitors, this value is the same for all monitors.
(Read and Write property)

24.3.22   LogPixelsY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of pixels per logical inch along the screen height.
Example:

dim g as Graphics = OpenPrinter
dim info as WindowsGraphicsInfoMBS = g.WindowsGraphicsInfoMBS

MsgBox "Printer resolution: "+str(info.LogPixelsX)+" x "+str(info.LogPixelsY)

Notes: In a system with multiple display monitors, this value is the same for all monitors.
(Read and Write property)

24.3.23   MakersCount as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of markers the device has.
Notes: (Read and Write property)

24.3.24   PenCount as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of device-specific pens.
Notes: (Read and Write property)

24.3.25   PhysicalHeight as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printing devices: the height of the physical page, in device units.
Notes: For example, a printer set to print at 600 dpi on 8.5-by-11-inch paper has a physical height value of 6600 device units. Note that the physical page is almost always greater than the printable area of the page, and never smaller.
(Read and Write property)

24.3.26   PhysicalOffsetX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printing devices: the distance from the left edge of the physical page to the left edge of the printable area, in device units.
Notes: For example, a printer set to print at 600 dpi on 8.5-by-11-inch paper, that cannot print on the leftmost 0.25-inch of paper, has a horizontal physical offset of 150 device units.
(Read and Write property)

24.3.27   PhysicalOffsetY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printing devices: the distance from the top edge of the physical page to the top edge of the printable area, in device units.
Notes: For example, a printer set to print at 600 dpi on 8.5-by-11-inch paper, that cannot print on the topmost 0.5-inch of paper, has a vertical physical offset of 300 device units.
(Read and Write property)

24.3.28   PhysicalWidth as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For printing devices: the width of the physical page, in device units.
Notes: For example, a printer set to print at 600 dpi on 8.5-x11-inch paper has a physical width value of 5100 device units. Note that the physical page is almost always greater than the printable area of the page, and never smaller.
(Read and Write property)

24.3.29 Planes as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of color planes.
Notes: (Read and Write property)

24.3.30 ResolutionX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Width, in pixels, of the screen; or for printers, the width, in pixels, of the printable area of the page.
Notes: (Read and Write property)

24.3.31 ResolutionY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Height, in raster lines, of the screen; or for printers, the height, in pixels, of the printable area of the page.
Notes: (Read and Write property)

24.3.32 ScalingFactorX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scaling factor for the x-axis of the printer.
Notes: (Read and Write property)

24.3.33 ScalingFactorY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scaling factor for the y-axis of the printer.
Notes: (Read and Write property)

24.3.34 SizeX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Width, in millimeters, of the physical screen.
Notes: (Read and Write property)

24.3.35   SizeY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Height, in millimeters, of the physical screen.
Notes: (Read and Write property)

24.3.36   Technology as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Device technology. It can be any one of the kTechnology* constants.
Notes: (Read and Write property)

24.3.37   VRefresh as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For display devices: the current vertical refresh rate of the device, in cycles per second (Hz).
Notes: (Read and Write property)

24.3.38   Constants

Technology Constants

Constant	Value	Description
kTechnologyCharStream	4	Character stream
kTechnologyDisplayFile	6	Display file
kTechnologyMetaFile	5	Metafile
kTechnologyPlotter	0	Vector plotter
kTechnologyRasterCamera	3	Raster camera
kTechnologyRasterDisplay	1	Raster display
kTechnologyRasterPrinter	2	Raster printer

24.4 class WindowsPageFormatMBS

24.4.1 class WindowsPageFormatMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a page format.
Example:

```
// change the printer name to your printer's name
dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)
dim formats(-1) as WindowsPageFormatMBS = w.GetPrinterFormats

for each p as WindowsPageFormatMBS in formats
MsgBox p.Name
next
```

Notes: This class wraps the FORM_INFO_1 structure. You can find more information here:
http://msdn.microsoft.com/en-us/library/dd144836(v=VS.85).aspx
Blog Entries

- MBS Real Studio Plugins, version 12.1pr1

24.4.2 Properties

24.4.3 DisplayName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The form's display name in the language specified by LangId.
Notes: Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.4 Flags as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The flags for this page format.
Notes: The following values are defined.

(Read and Write property)

Value	Meaning
FORM_USER	If this bit flag is set, the form has been defined by the user. Forms with this flag set are defined in the registry.
FORM_BUILTIN	If this bit-flag is set, the form is part of the spooler. Form definitions with this flag set do not appear in the registry.
FORM_PRINTER	If this bit flag is set, the form is associated with a certain printer, and its definition appears in the registry.

24.4.5   ImageableAreaBottom as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.6   ImageableAreaHeight as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.7   ImageableAreaLeft as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.8   ImageableAreaRight as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.9   ImageableAreaTop as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.10   ImageableAreaWidth as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The imageable area, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.11   Keyword as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A non-localizable string identifier of the form.
Notes: When passed to AddForm or SetForm, this gives the caller a means of identifying the form in all locales.
Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.12   LangId as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The language of the DisplayName.
Notes: Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.13   Mode as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The mode.
Notes: Value can be 1 (Windows 2000 and newer) or 2 (Windows Vista and newer).
(Read and Write property)

24.4.14   MuiDll as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Multilingual User Interface localized resource DLL that contains the localized display name.
Notes: Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.15   Name as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the form.
Notes: (Read and Write property)

24.4.16   ResourceId as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The resource ID of the form's display name in MuiDll.
Notes: Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.17   SizeHeight as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.18   SizeWidth as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width, in thousandths of millimeters, of the form.
Notes: (Read and Write property)

24.4.19   StringType as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies how a localized display name for the form is obtained at runtime.
Notes: The following values are defined. Only one can be set in any given call to AddForm or SetForm. Both STRING_MUIDLL and STRING_LANGPAIR can be set in the WindowsPageFormatMBS (s) returned by

GetForm or GetPrinterFormats.

Can be STRING_LANGPAIR, STRING_MUIDLL or STRING_NONE.
Only available in Windows Vista and newer and only valid if mode = 2.
(Read and Write property)

24.4.20   Constants

Constants

Constant	Value	Description
FORM_BUILTIN	1	One of the flag constants.
		If this bit-flag is set, the form is part of the spooler. Form definitions with this flag set do not appear in the registry.
FORM_PRINTER	2	One of the flag constants.
		If this bit flag is set, the form is associated with a certain printer, and its definition appears in the registry.
FORM_USER	0	One of the flag constants.
		If this bit flag is set, the form has been defined by the user. Forms with this flag set are defined in the registry.

String Type Constants

Constant	Value	Description
STRING_LANGPAIR	4	The display name and language ID are provided directly by DisplayName and the language is specified by LangId.
STRING_MUIDLL	2	The display name is extracted from the Multilingual User Interface localized resources DLL specified in MuiDll. The ID is in the ResourceId member.
STRING_NONE	1	There is no localized display name.

24.5   class WindowsPageSetupDialogMBS

24.5.1   class WindowsPageSetupDialogMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The class to run the Windows Page Setup dialog.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3
- MBS Real Studio Plugins, version 12.1pr7

24.5.2   Methods

24.5.3   Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The constructor.

24.5.4   GetDevNames(byref DriverName as string, byref DeviceName as string, byref OutputName as string, byref flags as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Queries the strings that identify the driver, device, and output port names for a printer.
Notes: Returns true if values are available and false if not.

DriverName: The file name (without the extension) of the device driver. On input, this string is used to determine the printer to display initially in the dialog box.
OutputName: The device name for the physical output medium (output port).
Flags: Indicates whether the strings here identify the default printer. This string is used to verify that the default printer has not changed since the last print operation. If any of the strings do not match, a warning message is displayed informing the user that the document may need to be reformatted. On output, the Flags member is changed only if the Print Setup dialog box was displayed and the user chose the OK button. The DN_DEFAULTPRN flag is used if the default printer was selected. If a specific printer is selected, the flag is not used. All other flags in this member are reserved for internal use by the dialog box procedure for the Print property sheet or Print dialog box.

24.5.5   PageSetupDialog as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Creates a Page Setup dialog box that enables the user to specify the attributes of a printed page.
Notes: These attributes include the paper size and source, the page orientation (portrait or landscape), and the width of the page margins.

If the user clicks the OK button, the return value is true. The members of the object indicate the user's selections.
If the user cancels or closes the Page Setup dialog box or an error occurs, the return value is false. To get extended error information, use the lasterror property.

Starting with Windows Vista, the PageSetupDialog does not contain the Printer button. To switch printer selection other functions.

24.5.6 SetDevNames(DriverName as string, DeviceName as string, Output-Name as string, flags as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Sets the strings that identify the driver, device, and output port names for a printer.
Notes: Returns true on success and false on failure.

DriverName: The file name (without the extension) of the device driver. On input, this string is used to determine the printer to display initially in the dialog box.
OutputName: The device name for the physical output medium (output port).
Flags: Indicates whether the strings here identify the default printer. This string is used to verify that the default printer has not changed since the last print operation. If any of the strings do not match, a warning message is displayed informing the user that the document may need to be reformatted. On output, the Flags member is changed only if the Print Setup dialog box was displayed and the user chose the OK button. The DN_DEFAULTPRN flag is used if the default printer was selected. If a specific printer is selected, the flag is not used. All other flags in this member are reserved for internal use by the dialog box procedure for the Print property sheet or Print dialog box.

24.5.7 Properties

24.5.8 DevMode as WindowsDeviceModeMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The device mode object.
Notes: On input, if an object is specified, the values in the corresponding DEVMODE structure are used to initialize the controls in the dialog box. On output, the dialog box sets DevMode to a new object that contains values specifying the user's selections. If the user's selections are not available, the dialog box sets DevMode to nil.

(Read and Write property)

24.5.9   Flags as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: A set of bit flags that you can use to initialize the Page Setup dialog box.
Notes: When the dialog box returns, it sets these flags to indicate the user's input.
See the PSD_* constants for possible values.
(Read and Write property)

24.5.10   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

24.5.11   MarginBottom as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The widths of the bottom margin.
Notes: If you set the PSD_MARGINS flag, rtMargin specifies the initial margin values. When PageSetupDlg returns, rtMargin contains the margin widths selected by the user. The PSD_INHUNDREDTH-SOFMILLIMETERS or PSD_INTHOUSANDTHSOFINCHES flag indicates the units of measurement.
(Read and Write property)

24.5.12   MarginLeft as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The widths of the left margin.
Notes: If you set the PSD_MARGINS flag, rtMargin specifies the initial margin values. When PageSetupDlg returns, rtMargin contains the margin widths selected by the user. The PSD_INHUNDREDTH-SOFMILLIMETERS or PSD_INTHOUSANDTHSOFINCHES flag indicates the units of measurement.
(Read and Write property)

24.5.13 MarginRight as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The widths of the right margin.
Notes: If you set the PSD_MARGINS flag, rtMargin specifies the initial margin values. When PageSetupDlg returns, rtMargin contains the margin widths selected by the user. The PSD_INHUNDREDTH-SOFMILLIMETERS or PSD_INTHOUSANDTHSOFINCHES flag indicates the units of measurement.
(Read and Write property)

24.5.14 MarginTop as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The widths of the top margin.
Notes: If you set the PSD_MARGINS flag, rtMargin specifies the initial margin values. When PageSetupDlg returns, rtMargin contains the margin widths selected by the user. The PSD_INHUNDREDTH-SOFMILLIMETERS or PSD_INTHOUSANDTHSOFINCHES flag indicates the units of measurement.
(Read and Write property)

24.5.15 MinMarginBottom as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The minimum allowable widths for the bottom margin.
Notes: The system ignores this member if the PSD_MINMARGINS flag is not set. These values must be less than or equal to the values specified in the Margin* member. The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.16 MinMarginLeft as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The minimum allowable widths for the left margin.
Notes: The system ignores this member if the PSD_MINMARGINS flag is not set. These values must be less than or equal to the values specified in the Margin* member. The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.17 MinMarginRight as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The minimum allowable widths for the right margin.
Notes: The system ignores this member if the PSD_MINMARGINS flag is not set. These values must be less than or equal to the values specified in the Margin* member. The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.18 MinMarginTop as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The minimum allowable widths for the top margin.
Notes: The system ignores this member if the PSD_MINMARGINS flag is not set. These values must be less than or equal to the values specified in the Margin* member. The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.19 PaperSizeX as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The dimensions of the paper selected by the user.
Notes: The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.20 PaperSizeY as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The dimensions of the paper selected by the user.
Notes: The PSD_INTHOUSANDTHSOFINCHES or PSD_INHUNDREDTHSOFMILLIMETERS flag indicates the units of measurement.
(Read and Write property)

24.5.21 Parent as Variant

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The window that owns the dialog box.
Notes: This member can be any valid window, or it can be nil if the dialog box has no owner.
Can reference a Window or DesktopWindow object.
(Read and Write property)

24.5.22   Constants

Constants

Constant	Value	Description
DN_DEFAULTPRN	1	The flag for default printer on the SetDevNames/GetDevNames flag parameter.

Flags

Constant	Value	Description
PSD_DEFAULTMINMARGINS	&h00000000	Sets the minimum values that the user can specify for th be the minimum margins allowed by the printer. This is flag is ignored if the PSD_MARGINS and PSD_MINMAR specified.
PSD_DISABLEMARGINS	&h00000010	Disables the margin controls, preventing the user from sett
PSD_DISABLEORIENTATION	&h00000100	Disables the orientation controls, preventing the user fron orientation.
PSD_DISABLEPAPER	&h00000200	Disables the paper controls, preventing the user from settin such as the paper size and source.
PSD_DISABLEPRINTER	&h00000020	Obsolete. Windows XP/2000: Disables the Printer button, preven invoking a dialog box that contains additional printer setup
PSD_INHUNDREDTHSOFMILLIMETERS	&h00000008	Indicates that hundredths of millimeters are the unit of mea gins and paper size. The values in the Margin, MinMarg members are in hundredths of millimeters. You can set to override the default unit of measurement for the user's function returns, the dialog box sets this flag to indicate th
PSD_INTHOUSANDTHSOFINCHES	&h00000004	Indicates that thousandths of inches are the unit of measu and paper size. The values in the Margin, MinMargin, and are in thousandths of inches. You can set this flag on in default unit of measurement for the user's locale. When th the dialog box sets this flag to indicate the units used.
PSD_MARGINS	&h00000002	Causes the system to use the values specified in the Marg initial widths for the left, top, right, and bottom margins. is not set, the system sets the initial widths to one inch for
PSD_MINMARGINS	&h00000001	Causes the system to use the values specified in the MinM the minimum allowable widths for the left, top, right, an The system prevents the user from entering a width that is ified minimum. If PSD_MINMARGINS is not specified, t minimum allowable widths to those allowed by the printer.
PSD_NONETWORKBUTTON	&h00200000	Hides and disables the Network button.
PSD_NOWARNING	&h00000080	Prevents the system from displaying a warning message whe printer.
PSD_RETURNDEFAULT	&h00000400	PageSetupDialog does not display the dialog box. Instead Names and DevMode members to objects that are initiali default printer. PageSetupDlg returns an error if either Dev is not nil.
PSD_SHOWHELP	&h00000800	Causes the dialog box to display the Help button. The hw must specify the window to receive the HELPMSGSTRING that the dialog box sends when the user clicks the Help but

24.6 class WindowsPrintDialogMBS

24.6.1 class WindowsPrintDialogMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The class for showing the windows print dialog.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3

- MBS Xojo Plugins, version 21.3pr2

- MBS Real Studio Plugins, version 12.1pr7

24.6.2 Methods

24.6.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The constructor.

24.6.4 GetDevNames(byref DriverName as string, byref DeviceName as string, byref OutputName as string, byref flags as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Queries the strings that identify the driver, device, and output port names for a printer.
Notes: Returns true if values are available and false if not.

DriverName: The file name (without the extension) of the device driver. On input, this string is used to determine the printer to display initially in the dialog box.
OutputName: The device name for the physical output medium (output port).
Flags: Indicates whether the strings here identify the default printer. This string is used to verify that the default printer has not changed since the last print operation. If any of the strings do not match, a warning message is displayed informing the user that the document may need to be reformatted. On output, the Flags member is changed only if the Print Setup dialog box was displayed and the user chose the OK button. The DN_DEFAULTPRN flag is used if the default printer was selected. If a specific printer is selected, the flag is not used. All other flags in this member are reserved for internal use by the dialog box procedure for the Print property sheet or Print dialog box.

24.6.5   getPageRange(index as Integer, byref fromPage as Integer, byref toPage as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Queries a page range entry.

24.6.6   PrintDialog as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Displays a Print Dialog Box or a Print Setup dialog box.
Notes: The Print dialog box enables the user to specify the properties of a particular print job.

If the user clicks the OK button, the return value is true. The members of the object indicate the user's selections.

If the user canceled or closed the Print or Printer Setup dialog box or an error occurred, the return value is zero. To get extended error information, use the lasterror property. If the user canceled or closed the dialog box, lasterror returns zero.

This is the older function. PrintDialogEx is the newer function.
see also
http://msdn.microsoft.com/en-us/library/windows/desktop/ms646940(v=vs.85).aspx

24.6.7   PrintDialogEx as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Displays a Print property sheet that enables the user to specify the properties of a particular print job.
Notes: Changed from Boolean to Integer in v21.3 to allow you to distinguish between different different results. Result is either 0 for OK or an COM error code.

A Print property sheet has a General page that contains controls similar to the Print dialog box. The property sheet can also have additional application-specific and driver-specific property pages as well as the General page.
If the function succeeds, the return value is S_OK (0) and the ResultAction member contains one of the following values.

Lasterror is set.
The values of DevMode and DevNames in WindowsPrintDialogMBS may change when they are passed into

Constant	Value	Description
PD_RESULT_APPLY	2	The user clicked the Apply button and later clicked the Cancel button. This indicates that the user wants to apply the changes made in the property sheet, but does not yet want to print. The WindowsPrintDialogMBS contains the information specified by the user at the time the Apply button was clicked.
PD_RESULT_CANCEL	0	The user clicked the Cancel button. The information in the WindowsPrintDialogMBS is unchanged.
PD_RESULT_PRINT	1	The user clicked the Print button. The WindowsPrintDialogMBS contains the information specified by the user.

PrintDialog. This is because these members are filled on both input and output.

If PD_RETURNDC is set but PD_USEDEVMODECOPIESANDCOLLATE flag is not set, the PrintDialog functions return incorrect number of copies. To get the correct number of copies, ensure that the calling application always uses PD_USEDEVMODECOPIESANDCOLLATE with PD_RETURNDC.

see also

http://msdn.microsoft.com/en-us/library/windows/desktop/ms646942(v=vs.85).aspx

24.6.8 SetDevNames(DriverName as string, DeviceName as string, Output- Name as string, flags as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Sets the strings that identify the driver, device, and output port names for a printer.
Notes: Returns true on success and false on failure.

DriverName: The file name (without the extension) of the device driver. On input, this string is used to determine the printer to display initially in the dialog box.

OutputName: The device name for the physical output medium (output port).

Flags: Indicates whether the strings here identify the default printer. This string is used to verify that the default printer has not changed since the last print operation. If any of the strings do not match, a warning message is displayed informing the user that the document may need to be reformatted. On output, the Flags member is changed only if the Print Setup dialog box was displayed and the user chose the OK button. The DN_DEFAULTPRN flag is used if the default printer was selected. If a specific printer is selected, the flag is not used. All other flags in this member are reserved for internal use by the dialog box procedure for the Print property sheet or Print dialog box.

24.6.9 setPageRange(index as Integer, fromPage as Integer, toPage as Integer)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Sets a page range entry.
Notes: Index from 0 to 19.

24.6.10   Properties

24.6.11   Copies as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Contains the initial number of copies for the Copies edit control if DevMode is nil; otherwise, the Copies member of the DEVMODE contains the initial value.
Notes: When PrintDlgEx returns, nCopies contains the actual number of copies the application must print. This value depends on whether the application or the printer driver is responsible for printing multiple copies. If the PD_USEDEVMODECOPIESANDCOLLATE flag is set in the Flags member, nCopies is always 1 on return, and the printer driver is responsible for printing multiple copies. If the flag is not set, the application is responsible for printing the number of copies specified by nCopies. For more information, see the description of the PD_USEDEVMODECOPIESANDCOLLATE flag.
(Read and Write property)

24.6.12   DC as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The device context or an information context, depending on whether the Flags member specifies the PD_RETURNDC or PC_RETURNIC flag.
Notes: If neither flag is specified, the value of this member is undefined. If both flags are specified, PD_RETURNDC has priority.
(Read and Write property)

24.6.13   DevMode as WindowsDeviceModeMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The device mode object.
Notes: On input, if an object is specified, the values in the corresponding DEVMODE structure are used to initialize the controls in the dialog box. On output, the dialog box sets DevMode to a new object that contains values specifying the user's selections. If the user's selections are not available, the dialog box sets DevMode to nil.
(Read and Write property)

24.6.14   ExclusionFlags as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: A set of bit flags that can exclude items from the printer driver property pages in the Print property sheet.

Notes: This value is used only if the PD_EXCLUSIONFLAGS flag is set in the Flags member. Exclusion flags should be used only if the item to be excluded will be included on either the General page or on an application-defined page in the Print property sheet. This member can specify the following flag.

PD_EXCL_COPIESANDCOLLATE: Excludes the Copies and Collate controls from the printer driver property pages in a Print property sheet. This flag should always be set when the application uses the default Copies and Collate controls provided by the lower portion of the General page of the Print property sheet.

(Read and Write property)

24.6.15   Flags as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: A set of bit flags that you can use to initialize the Print property sheet.

Notes: When the PrintDialog function returns, it sets these flags to indicate the user's input. This member can be one or more of the following values.

To ensure that PrintDialog returns the correct values in the Copies and Collate members of the DeviceMode, set PD_RETURNDC = TRUE and PD_USEDEVMODECOPIESANDCOLLATE = TRUE. In so doing, the Copies member of the PRINTDLG structure is always 1 and PD_COLLATE is always false.

To ensure that PrintDialog returns the correct values in nCopies and PD_COLLATE, set PD_RETURNDC = TRUE and PD_USEDEVMODECOPIESANDCOLLATE = FALSE. In so doing, dmCopies is always 1 and dmCollate is always FALSE.

Starting with Windows Vista, when you call PrintDlg or PrintDlgEx with PD_RETURNDC set to TRUE and PD_USEDEVMODECOPIESANDCOLLATE set to FALSE, the PrintDlg or PrintDlgEx function sets the number of copies in the nCopies member of the PRINTDLG structure, and it sets the number of copies in the structure represented by the hDC member of the PRINTDLG structure.

When making calls to GDI, you must ignore the value of nCopies, consider the value as 1, and use the returned hDC to avoid printing duplicate copies.

(Read and Write property)

24.6.16   FromPage as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The initial value for the starting page edit control.

Notes: When PrintDialog returns, nFromPage is the starting page specified by the user. If the Pages radio button is selected when the user clicks the Okay button, PrintDialog sets the PD_PAGENUMS flag and does not return until the user enters a starting page value that is within the minimum to maximum page range.

If the input value for either FromPage or nToPage is outside the minimum/maximum range, PrintDlg returns an error only if the PD_PAGENUMS flag is specified; otherwise, it displays the dialog box but changes the out-of-range value to the minimum or maximum value.

This is only for PrintDialog, not for PrintDialogEx.

(Read and Write property)

24.6.17   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: Typical errors:
E_OUTOFMEMORY = &h8007000E // Insufficient memory.
E_INVALIDARG = &h80070057 // One or more arguments are invalid.
E_POINTER = &h80000005 // Invalid pointer.
E_HANDLE = &h80000006 // Invalid handle.
E_FAIL = &h80000008 // Unspecified error.
(Read and Write property)

24.6.18   MaxPage as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The maximum value for the page ranges specified in the Pages edit control.
Notes: If the PD_NOPAGENUMS flag is specified, this value is not valid.
(Read and Write property)

24.6.19   MaxPageRanges as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The size, in array elements, of the PageRanges buffer.
Notes: This value indicates the maximum number of page ranges that can be stored in the array. If the PD_NOPAGENUMS flag is specified, this value is not valid. If the PD_NOPAGENUMS flag is not specified, this value must be greater than zero.
The plugin supports up to 20 page ranges.
(Read and Write property)

24.6.20   MinPage as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The minimum value for the page ranges specified in the Pages edit control.
Notes: If the PD_NOPAGENUMS flag is specified, this value is not valid.
(Read and Write property)

24.6.21 PageRanges as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: On input, set this member to the initial number of page ranges specified.
Notes: When the PrintDialog function returns, PageRanges indicates the number of user-specified page ranges. If the PD_NOPAGENUMS flag is specified, this value is not valid. Use getPageRange to query page ranges or setPageRange to set them.
(Read and Write property)

24.6.22 Parent as Variant

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The window that owns the dialog box.
Notes: The window that owns the property sheet. This member must be a valid window handle; it cannot be nil.
Can reference a Window or DesktopWindow object.
(Read and Write property)

24.6.23 ResultAction as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: If the PrintDlgEx function returns S_OK (0), ResultAction contains the outcome of the dialog.
Notes: If PrintDialog returns an error, this member should be ignored. The ResultAction member can be one of the following values:

PD_RESULT_APPLY	The user clicked the Apply button and later clicked the Cancel button. This indicates that the user wants to apply the changes made in the property sheet, but does not want to print yet. The PRINTDLGEX structure contains the information specified by the user at the time the Apply button was clicked.
PD_RESULT_CANCEL	The user clicked the Cancel button. The information in the PRINTDLGEX structure is unchanged.
PD_RESULT_PRINT	The user clicked the Print button. The PRINTDLGEX structure contains the information specified by the user.

(Read and Write property)

24.6.24 StartPanel as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The property page that is initially displayed.
Notes: To display the General page, specify START_PAGE_GENERAL. For consistency, it is recommended that the property sheet always be started on the General page.
The plugin sets START_PAGE_GENERAL for default value.
(Read and Write property)

24.6.25 ToPage as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The initial value for the ending page edit control.
Notes: When PrintDialog returns, ToPage is the ending page specified by the user. If the Pages radio button is selected when the use clicks the Okay button, PrintDialog sets the PD_PAGENUMS flag and does not return until the user enters an ending page value that is within the minimum to maximum page range. This is only for PrintDialog, not for PrintDialogEx.
(Read and Write property)

24.6.26 Constants

Constants

Constant	Value	Description
START_PANEL_GENERAL	&hffffffff	Special value for StartPanel property to select general panel.

Flag Constants

Constant	Value	Description
PD_ALLPAGES	0	The default flag that indicates that the All radio button i This flag is used as a placeholder to indicate that the PD_SELECTION, and PD_CURRENTPAGE flags are
PD_COLLATE	&h00000010	If this flag is set, the Collate check box is selected. If this flag is set when the PrintDlgEx function returns, simulate collation of multiple copies. For more information of the PD_USEDEVMODECOPIESANDCOLLATE flag See PD_NOPAGENUMS.
PD_CURRENTPAGE	&h00400000	If this flag is set, the Current Page radio button is sele PD_PAGENUMS, PD_SELECTION, or PD_CURREN the All radio button is selected.
PD_DISABLEPRINTTOFILE	&h00080000	Disables the Print to File check box.
PD_EXCLUSIONFLAGS	&h01000000	Indicates that the ExclusionFlags member identifies items the printer driver property pages. If this flag is not set, items will be excluded by default fr property pages. The exclusions prevent the duplication General page, any application-specified pages, and the pr
PD_EXCL_COPIESANDCOLLATE	&h00008100	Excludes the Copies and Collate controls from the printer in a Print property sheet. This flag should always be set uses the default Copies and Collate controls provided by the General page of the Print property sheet.
PD_HIDEPRINTTOFILE	&h00100000	Hides the Print to File check box.
PD_NOCURRENTPAGE	&h00800000	Disables the Current Page radio button.
PD_NONETWORKBUTTON	&h00200000	Hides and disables the Network button.
PD_NOPAGENUMS	8	Disables the Pages radio button and the associated edit c Also, it causes the Collate check box to appear in the dia
PD_NOSELECTION	4	Disables the Selection radio button.
PD_NOWARNING	&h00000080	Prevents the warning message from being displayed when
PD_PAGENUMS	2	If this flag is set, the Pages radio button is selected. If r GENUMS, PD_SELECTION, or PD_CURRENTPAGE radio button is selected. If this flag is set when the Prin turns, the PageRanges member indicates the page ranges s
PD_PRINTSETUP	&h00000040	Causes the system to display the Print Setup dialog box r dialog box.
PD_PRINTTOFILE	&h00000020	If this flag is set, the Print to File check box is selected. If this flag is set when PrintDialog returns, the Output contains the string "FILE:". When you call the StartD the printing operation, specify this "FILE:" string in th member of the DOCINFO structure. Specifying this str subsystem to query the user for the name of the output f StartDoc is available in our plugins through StartDocPri dPrintJobMBS class.
PD_RETURNDC	&h00000100	Causes PrintDialog to return a device context matching t made in the property sheet. The device context is returned in DC property.
PD_RETURNDEFAULT	&h00000400	If this flag is set, the PrintDialog function does not displa Instead, it sets the DevNames and DevMode members t system default printer. Both DevNames and DevMode PrintDialog returns an error.
PD_RETURNIC	&h00000200	Similar to the PD_RETURNDC flag, except this flag ret context rather than a device context. If neither PD_RETURNDC nor PD_RETURNIC is spe fined on output.
PD_SELECTION	1	If this flag is set, the Selection radio button is selected. If none of the PD_PAGENUMS, PD_SELECTION, or PD flags is set, the All radio button is selected.
PD_SHOWHELP	&h00000800	Causes the dialog box to display the Help button.
PD_USEDEVMODECOPIES	&h00040000	Same as PD_USEDEVMODECOPIESANDCOLLATE.

Result Constants

Constant	Value	Description
PD_RESULT_APPLY	2	The user clicked the Apply button and later clicked the Cancel button. This indicates that the user wants to apply the changes made in the property sheet, but does not yet want to print. The WindowsPrintDialogMBS contains the information specified by the user at the time the Apply button was clicked.
PD_RESULT_CANCEL	0	The user clicked the Cancel button. The information in the WindowsPrintDialogMBS is unchanged.
PD_RESULT_PRINT	1	The user clicked the Print button. The WindowsPrintDialogMBS contains the information specified by the user.

24.7 class WindowsPrinterInfoMBS

24.7.1 class WindowsPrinterInfoMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for details on printer information.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName
next

Notes: This class wraps the PRINTER_INFO_2 structure on windows which you find documented here:
http://msdn.microsoft.com/en-us/library/dd162845(VS.85).aspx
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Print to Printer in Xojo on Windows

- MBS Xojo / Real Studio Plugins, version 15.2pr6

- MBS Real Studio Plugins, version 12.1pr6

24.7.2 Methods

24.7.3 Constructor

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

24.7.4 LocalPrinters as WindowsPrinterInfoMBS()

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the printers as an array.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName

next

Notes: On any error the array will be empty.

24.7.5   OpenPrinter(admin as boolean = false) as WindowsPrinterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens the printer so you can operate on it.
Example:

dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name)
MsgBox p.PrinterName

Notes: Admin: whether you want to get permissions to administrate.
Returns nil on any error.

24.7.6   Printers(flags as Integer, Name as Variant = nil) as WindowsPrinterIn- foMBS()

Plugin Version: 15.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Printers function enumerates available printers, print servers, domains, or print providers.
Notes: If you call this with flags = kPrinterFlagsLocal + kPrinterFlagsConnections and name = nil, you get same as LocalPrinters function.

see also
https://msdn.microsoft.com/en-us/library/windows/desktop/dd162692(v=vs.85).aspx

24.7.7   Properties

24.7.8   AttributeFlags as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the printer attributes.
Notes: This member can be one of the following values:

PRINTER_ATTRIBUTE_QUEUED	&h00000001	
PRINTER_ATTRIBUTE_DIRECT	&h00000002	
PRINTER_ATTRIBUTE_DEFAULT	&h00000004	(Windows 95)
PRINTER_ATTRIBUTE_SHARED	&h00000008	
PRINTER_ATTRIBUTE_NETWORK	&h00000010	
PRINTER_ATTRIBUTE_HIDDEN	&h00000020	
PRINTER_ATTRIBUTE_LOCAL	&h00000040	
PRINTER_ATTRIBUTE_ENABLE_DEVQ	&h00000080	
PRINTER_ATTRIBUTE_KEEPPRINTEDJOBS	&h00000100	
PRINTER_ATTRIBUTE_DO_COMPLETE_FIRST	&h00000200	
PRINTER_ATTRIBUTE_WORK_OFFLINE	&h00000400	(Windows 95)
PRINTER_ATTRIBUTE_ENABLE_BIDI	&h00000800	(Windows 95)
PRINTER_ATTRIBUTE_RAW_ONLY	&h00001000	
PRINTER_ATTRIBUTE_PUBLISHED	&h00002000	(Windows 2000) Indicates whether the printer is published in the directory service.

Renamed Attributes parameter to AttributeFlags in plugin version 8.2.
(Read and Write property)

24.7.9   AveragePPM as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the average number of pages per minute that have been printed on the printer.
Notes: (Read and Write property)

24.7.10   Comment as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that provides a brief description of the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName+": "+p.Comment
next

Notes: (Read and Write property)

24.7.11   CountJobs as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the number of print jobs that have been queued for the printer.
Notes: (Read and Write property)

24.7.12 Datatype as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the data type used to record the print job.
Notes: (Read and Write property)

24.7.13 DefaultPriority as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the default priority value assigned to each print job.
Notes: (Read and Write property)

24.7.14 DevMode as WindowsDeviceModeMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WindowsDeviceModeMBS object that defines default printer data such as the paper orientation and the resolution.
Notes: (Read and Write property)

24.7.15 DriverName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the name of the printer driver.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName+": "+p.DriverName
next

Notes: (Read and Write property)

24.7.16   Location as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the physical location of the printer.
Notes: for example, "Bldg. 38, Room 1164".
(Read and Write property)

24.7.17   Parameters as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the default print-processor parameters.
Notes: (Read and Write property)

24.7.18   PortName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that identifies the port(s) used to transmit data to the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName+": "+p.PortName
next

Notes: If a printer is connected to more than one port, the names of each port must be separated by commas (for example, "LPT1:,LPT2:,LPT3:").
(Read and Write property)

24.7.19   PrinterName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the name of the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers

MsgBox p.PrinterName
next

Notes: (Read and Write property)

24.7.20   PrintProcessor as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the name of the print processor used by the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrintProcessor
next

Notes: (Read and Write property)

24.7.21   Priority as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a priority value that the spooler uses to route print jobs.
Notes: Possible values:

NO_PRIORITY	0
MAX_PRIORITY	99
MIN_PRIORITY	1
DEF_PRIORITY	1

(Read and Write property)

24.7.22   SeparatorPageFile as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the name of the file used to create the separator page.
Notes: This page is used to separate print jobs sent to the printer.
(Read and Write property)

24.7.23 ServerName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string identifying the server that controls the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName+": "+p.ServerName
next

Notes: If this string is "", the printer is controlled locally.
(Read and Write property)

24.7.24 ShareName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that identifies the sharepoint for the printer.
Example:

dim printers(-1) as WindowsPrinterInfoMBS = WindowsPrinterInfoMBS.LocalPrinters

for each p as WindowsPrinterInfoMBS in printers
MsgBox p.PrinterName+": "+p.ShareName
next

Notes: This string is used only if the PRINTER_ATTRIBUTE_SHARED constant was set for the Attributes member.
(Read and Write property)

24.7.25 StartTime as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the earliest time at which the printer will print a job.
Notes: This value is expressed as minutes elapsed since 12:00 A.M. GMT (Greenwich Mean Time).
(Read and Write property)

24.7.26   Status as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the printer status.
Notes: This member can be one of the following values:

PRINTER_STATUS_PAUSED	&h00000001
PRINTER_STATUS_ERROR	&h00000002
PRINTER_STATUS_PENDING_DELETION	&h00000004
PRINTER_STATUS_PAPER_JAM	&h00000008
PRINTER_STATUS_PAPER_OUT	&h00000010
PRINTER_STATUS_MANUAL_FEED	&h00000020
PRINTER_STATUS_PAPER_PROBLEM	&h00000040
PRINTER_STATUS_OFFLINE	&h00000080
PRINTER_STATUS_IO_ACTIVE	&h00000100
PRINTER_STATUS_BUSY	&h00000200
PRINTER_STATUS_PRINTING	&h00000400
PRINTER_STATUS_OUTPUT_BIN_FULL	&h00000800
PRINTER_STATUS_NOT_AVAILABLE	&h00001000
PRINTER_STATUS_WAITING	&h00002000
PRINTER_STATUS_PROCESSING	&h00004000
PRINTER_STATUS_INITIALIZING	&h00008000
PRINTER_STATUS_WARMING_UP	&h00010000
PRINTER_STATUS_TONER_LOW	&h00020000
PRINTER_STATUS_NO_TONER	&h00040000
PRINTER_STATUS_PAGE_PUNT	&h00080000
PRINTER_STATUS_USER_INTERVENTION	&h00100000
PRINTER_STATUS_OUT_OF_MEMORY	&h00200000
PRINTER_STATUS_DOOR_OPEN	&h00400000
PRINTER_STATUS_SERVER_UNKNOWN	&h00800000
PRINTER_STATUS_POWER_SAVE	&h01000000

Windows 95: The PRINTER_STATUS_PAGE_PUNT value specifies that the page is being "punted" (that is, not printed) because it is too complex for the printer to print.
(Read and Write property)

24.7.27   UntilTime as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the latest time at which the printer will print a job.
Notes: This value is expressed as minutes elapsed since 12:00 A.M. GMT (Greenwich Mean Time).
Status

(Read and Write property)

24.7.28 Constants

Printer Enumeration Flags

Constant	Value	Description
kPrinterFlagsConnections	4	The function enumerates the list of printers to which the user has made previous connections.
kPrinterFlagsDefault	1	Default printer.
kPrinterFlagsLocal	2	If the kPrinterFlagsName flag is not also passed, the function ignores the Name parameter, and enumerates the locally installed printers. If kPrinterFlagsName is also passed, the function enumerates the local printers on Name.
kPrinterFlagsName	8	The function enumerates the printer identified by Name. This can be a server, a domain, or a print provider. If Name is nil, the function enumerates available print providers.
kPrinterFlagsNetwork	64	Network printers?
kPrinterFlagsRemote	16	Remote (non Local) printers?
kPrinterFlagsShared	32	The function enumerates printers that have the shared attribute. Cannot be used in isolation; use an OR operation to combine with another constants.

24.8   class WindowsPrinterJobMBS

24.8.1   class WindowsPrinterJobMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for Windows print job information.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr4

24.8.2   Properties

24.8.3   Datatype as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The type of data used to record the print job.
Notes: (Read and Write property)

24.8.4   DevMode as WindowsDeviceModeMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device-initialization and environment data for the printer driver.
Notes: (Read and Write property)

24.8.5   Document as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the user who owns the print job.
Notes: (Read and Write property)

24.8.6   DriverName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the printer driver that should be used to process the print job.
Notes: (Read and Write property)

24.8.7 JobID as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A job identifier value.
Notes: (Read and Write property)

24.8.8 MachineName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the machine that created the print job.
Notes: (Read and Write property)

24.8.9 NotifyName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the user who should be notified when the job has been printed or when an error occurs while printing the job.
Notes: (Read and Write property)

24.8.10 PagesPrinted as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of pages that have printed.
Notes: This value may be zero if the print job does not contain page delimiting information. (Read and Write property)

24.8.11 Parameters as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Print-processor parameters.
Notes: (Read and Write property)

24.8.12 Position as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The job's position in the print queue.
Notes: (Read and Write property)

24.8.13   PrinterName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the printer for which the job is spooled.
Notes: (Read and Write property)

24.8.14   PrintProcessor as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the print processor that should be used to print the job.
Notes: (Read and Write property)

24.8.15   Priority as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The job priority.
Notes: This member can be one of the following values or in the range between 1 through 99 (kPriorityMin through kPriorityMax).
(Read and Write property)

24.8.16   Size as Int64

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size, in bytes, of the job.
Notes: (Read and Write property)

24.8.17   StartTime as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The earliest time that the job can be printed.
Notes: (Read and Write property)

24.8.18 Status as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The job status.
Notes: Check kJobStatus* constants for possible values.
(Read and Write property)

24.8.19 StatusString as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The status of the print job.
Notes: This member should be checked prior to Status and, if StatusString is "", the status is defined by the contents of the Status member.
(Read and Write property)

24.8.20 Submitted as Date

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time when the job was submitted.
Notes: This time value is in Universal Time Coordinate (UTC) format.
(Read and Write property)

24.8.21 Time as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The total time, in milliseconds, that has elapsed since the job began printing.
Notes: (Read and Write property)

24.8.22 TotalPages as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of pages required for the job.
Notes: This value may be zero if the print job does not contain page delimiting information.
(Read and Write property)

24.8.23   UntilTime as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The latest time that the job can be printed.
Notes: (Read and Write property)

24.8.24   UserName as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the user who owns the print job.
Notes: (Read and Write property)

24.8.25   Constants

Job Status Constants

Constant	Value	Description
kJobStatusBlockedDeviceQueue	&h200	The driver cannot print the job.
kJobStatusComplete	&h1000	The job is sent to the printer, but may not be printed yet. (Windows XP and later)
kJobStatusDeleted	&h100	Job has been deleted.
kJobStatusDeleting	4	Job is being deleted.
kJobStatusError	2	An error is associated with the job.
kJobStatusOffline	&h20	Printer is offline.
kJobStatusPaperOut	&h40	Printer is out of paper.
kJobStatusPaused	1	Job is paused.
kJobStatusPrinted	&h80	Job has printed.
kJobStatusPrinting	&h10	Job is printing.
kJobStatusRenderingLocally	&h4000	Rendering.
kJobStatusRestart	&h800	Job has been restarted.
kJobStatusRetained	&h2000	The job has been retained in the print queue following printing. (Windows XP and later)
kJobStatusSpooling	8	Job is spooling.
kJobStatusUserIntervention	&h400	Printer has an error that requires the user to do something.

Priority Constants

Constant	Value	Description
kPriorityDefault	1	Default priority.
kPriorityMax	99	Maximum priority.
kPriorityMin	1	Minimum priority.
kPriorityNo	0	No priority.

24.9 class WindowsPrinterMBS

24.9.1 class WindowsPrinterMBS

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for an open connection to a windows printer.
Notes: Objects of this class can operate on printers.
Some operations need administrative permissions. If you don't have them, lasterror is set to 5.
Blog Entries

- MonkeyBread Software releases MBS Real Studio plug-ins in version 12.1

- MBS Real Studio Plugins, version 12.1pr7

- MBS Real Studio Plugins, version 12.1pr6

- More on printing on Windows

- MBS Real Studio Plugins, version 12.1pr4

- MBS Real Studio Plugins, version 12.1pr1

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr2

24.9.2 Methods

24.9.3 AddForm(form as WindowsPageFormatMBS) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The AddForm function adds a form to the list of available forms that can be selected for the specified printer.
Example:

```
// query default printer
dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)

dim form as new WindowsPageFormatMBS

form.Mode = 1
form.Name = "MyTest"
form.DisplayName = "MyTest"
form.Flags = 0
form.SizeHeight = 100000
form.SizeWidth = 100000
form.ImageableAreaLeft = 0
```

form.ImageableAreaTop = 0
form.ImageableAreaRight = 100000
form.ImageableAreaBottom = 100000

if w.AddForm(form) then
MsgBox "OK"
else
MsgBox "Failed"
end if

Notes: Form.Mode gives level of the data. Must be either 1 (old version) or 2 (newer version).

Form: The form details.

Returns true if the function succeeds and false if the function fails.

Note: This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

An application can determine which forms are available for a printer by calling the GetPrinterFormats function.
If form points to a Mode 2 form, then AddForm will fail if either a form with the specified name already exists or the Keyword value already exists.

24.9.4 AdvancedDocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDevMode as WindowsDeviceModeMBS, parent as DesktopWindow) as integer

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

Function: Displays a printer-configuration dialog box for the specified printer, allowing the user to configure that printer.
Notes: This function is a special case of the DocumentProperties function.

parent: The parent window of the printer-configuration dialog box.

OutputDevMode: The device mode structure that will contain the configuration data specified by the user.
InputDevMode: The device mode structure that contains the configuration data used to initialize the con-

trols of the printer-configuration dialog box.

If the DocumentProperties function with these parameters is successful, the return value of AdvancedDocumentProperties is 1. Otherwise, the return value is zero.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

This function can only display the printer-configuration dialog box so a user can configure it. For more control, use DocumentProperties.
See also:

24.9.5 AdvancedDocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDevMode as WindowsDeviceModeMBS, parent as window = nil) as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Displays a printer-configuration dialog box for the specified printer, allowing the user to configure that printer.
Notes: This function is a special case of the DocumentProperties function.

parent: The parent window of the printer-configuration dialog box.

OutputDevMode: The device mode structure that will contain the configuration data specified by the user.
InputDevMode: The device mode structure that contains the configuration data used to initialize the controls of the printer-configuration dialog box.

If the DocumentProperties function with these parameters is successful, the return value of AdvancedDocumentProperties is 1. Otherwise, the return value is zero.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

This function can only display the printer-configuration dialog box so a user can configure it. For more

control, use DocumentProperties.
See also:

- 24.9.4 AdvancedDocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDev-
  Mode as WindowsDeviceModeMBS, parent as DesktopWindow) as integer   846

24.9.6   AllJobs as WindowsPrinterJobMBS()

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all jobs for the printer.
Example:

dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)
dim jobs() as WindowsPrinterJobMBS = w.AllJobs
MsgBox str(UBound(jobs)+1)+" jobs"

Notes: PrinterName property must be set, so the function knows which printer.

24.9.7   CanPrinterPassThroughPostScript as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this printer can pass through postscript data.
Notes: Returns true for Postscript printers and false for other printers.

For more information on this function, visit this website:
http://msdn.microsoft.com/en-us/library/dd162831(v=VS.85).aspx

24.9.8   ChangePrinterSettings(value as WindowsDeviceModeMBS, Mode as In-
teger=2) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Changes printer settings.
Notes: This function opens the printer, queries current settings using GetPrinter, applies all your changes,
calls DocumentProperties to verify those settings are valid for the printer and uses SetPrinter to apply the
changes.
To indicate which fiels in the WindowsDeviceModeMBS object are set by you, use the fields integer value
with a combination of the DM_* constants.
Returns true on success and false on failure.

Mode can be 2 (general printer information), 8 (global settings) or 9 (user settings).

For more information on the GetPrinter function, check this website:
http://msdn.microsoft.com/en-us/library/dd144911(VS.85).aspx

For more information on the SetPrinter function, check this website:
http://msdn.microsoft.com/en-us/library/dd145082(VS.85).aspx

For more information on the DocumentProperties function, check this website:
http://msdn.microsoft.com/en-us/library/dd183576(VS.85).aspx

Even if you change printer settings, other parts of the application, e.g. Xojo framework may not care.

24.9.9 ConfigurePort(name as string = "", parent as window = nil, PortName as string = "") as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Displays the port-configuration dialog box for a port on the specified server.
Notes: Name: The name of the server on which the specified port exists. If this parameter is "", the port is local.
parent: The parent window of the port-configuration dialog box.
PortName: The name of the port to be configured.

Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

Before calling the ConfigurePort function, an application should call the EnumPorts function to determine valid port names.
See also:

* 24.9.10 ConfigurePort(name as string, parent as DesktopWindow, PortName as string) as boolean 849

24.9.10 ConfigurePort(name as string, parent as DesktopWindow, PortName as string) as boolean

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

Function: Displays the port-configuration dialog box for a port on the specified server.
Notes: Name: The name of the server on which the specified port exists. If this parameter is "", the port is local.
parent: The parent window of the port-configuration dialog box.
PortName: The name of the port to be configured.

Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

Before calling the ConfigurePort function, an application should call the EnumPorts function to determine valid port names.
See also:

- 24.9.9 ConfigurePort(name as string = "", parent as window = nil, PortName as string = "") as boolean   849

24.9.11   ConnectToPrinterDialog(parent as DesktopWindow) as boolean

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

Function: Displays a dialog box that lets users browse and connect to printers on a network.
Notes: If the user selects a printer, the function attempts to create a connection to it; if a suitable driver is not installed on the server, the user is given the option of creating a printer locally.

parent: Specifies the parent window of the dialog box.

If the function succeeds and the user selects a printer, this object points to the new printer and return value is true.
If the function fails, or the user cancels the dialog box without selecting a printer, the return value is true.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

The ConnectToPrinterDialog function attempts to create a connection to the selected printer. However, if the server on which the printer resides does not have a suitable driver installed, the function offers the user the option of creating a printer locally.

An application should call DeletePrinter to delete a local printer. An application should call DeletePrinter-Connection to delete a connection to a printer.
See also:

24.9.12   ConnectToPrinterDialog(parent as window = nil) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: Displays a dialog box that lets users browse and connect to printers on a network.
Notes: If the user selects a printer, the function attempts to create a connection to it; if a suitable driver is not installed on the server, the user is given the option of creating a printer locally.

parent: Specifies the parent window of the dialog box.

If the function succeeds and the user selects a printer, this object points to the new printer and return value is true.
If the function fails, or the user cancels the dialog box without selecting a printer, the return value is true.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

The ConnectToPrinterDialog function attempts to create a connection to the selected printer. However, if the server on which the printer resides does not have a suitable driver installed, the function offers the user the option of creating a printer locally.

An application should call DeletePrinter to delete a local printer. An application should call DeletePrinter-Connection to delete a connection to a printer.
See also:

24.9.13   Constructor(PrinterName as string, admin as boolean = false)

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens the printer so you can operate on it.
Notes: Admin: whether you want to get permissions to administrate.
Raises exception if printer can't be opened. Use OpenPrinter function to avoid exception.

24.9.14   DeleteForm(name as string) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DeleteForm function removes a form name from the list of supported forms.
Notes: name: The form name to be removed.

Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

DeleteForm can only delete form names that were added by using the AddForm function.

24.9.15   DeleteJob(JobID as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Delete the print job.
Notes: Returns true on success and false on failure.

24.9.16   DeletePrinter as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes the specified printer object.
Notes: Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

If there are print jobs remaining to be processed for the specified printer, DeletePrinter marks the printer for pending deletion, and then deletes it when all the print jobs have been printed. No print jobs can be added to a printer that is marked for pending deletion.
A printer marked for pending deletion cannot be held, but its print jobs can be held, resumed, and restarted. If the printer is held and there are jobs for the printer, DeletePrinter fails with ERROR_ACCESS_DENIED.

Lasterror is set.

24.9.17 DeletePrinterConnection(name as string) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes a connection to a printer that was established by a call to AddPrinterConnection or ConnectToPrinterDialog.
Notes: name: The name of the printer connection to delete.
Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

The DeletePrinterConnection function does not delete any printer driver files that were copied to the server to which the printer is attached.

24.9.18 DocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDevMode as WindowsDeviceModeMBS, Prompt as boolean = false, parent as window = nil) as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The DocumentProperties function retrieves or modifies printer initialization information or displays a printer-configuration property sheet for the specified printer.
Example:

```
// get default printer
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(WindowsPrinterMBS.GetDefaultPrinter)

// ask for printer settings
dim s as new WindowsDeviceModeMBS
dim n as Integer = p.DocumentProperties(nil, s, true, window1)
```

Notes: Printername property must be set.

parent: The parent window of the printer-configuration property sheet. Can be nil.
Prompt: Whether to show dialog.

OutputDevMode: Devmode variable that receives the printer configuration data specified by the user.
InputDevMode: Optional input data which the operating system uses to initialize the property sheet controls.
Prompt: Whether the function presents the printer driver's Print Setup property sheet and then changes the settings in the printer's DeviceMode data structure to those values specified by the user.

If you provide object for InputDevMode, the function will read settings from there. If you don't pass an object (nil) or you pass incomplete settings, default settings are used to complete.
If prompt is true, the dialog is shown and user can change settings.
On the end, the current new data is stored in OutputDevMode where you can find it.

If the function displays the property sheet, the return value is either IDOK (1) or IDCANCEL (2), depending on which button the user selects.
If the function does not display the property sheet and is successful, the return value is IDOK (1).
If the function fails, the return value is less than zero.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation-factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd183576(v=vs.85).aspx
See also:

24.9.19   DocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDevMode as WindowsDeviceModeMBS, Prompt as boolean, parent as DesktopWindow) as integer

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

Function: The DocumentProperties function retrieves or modifies printer initialization information or displays a printer-configuration property sheet for the specified printer.
Example:

```
// get default printer
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(WindowsPrinterMBS.GetDefaultPrinter)

// ask for printer settings
dim s as new WindowsDeviceModeMBS
dim n as Integer = p.DocumentProperties(nil, s, true, window1)
```

Notes: Printername property must be set.

parent: The parent window of the printer-configuration property sheet. Can be nil.
Prompt: Whether to show dialog.

OutputDevMode: Devmode variable that receives the printer configuration data specified by the user.
InputDevMode: Optional input data which the operating system uses to initialize the property sheet controls.
Prompt: Whether the function presents the printer driver's Print Setup property sheet and then changes the settings in the printer's DeviceMode data structure to those values specified by the user.

If you provide object for InputDevMode, the function will read settings from there. If you don't pass an object (nil) or you pass incomplete settings, default settings are used to complete.
If prompt is true, the dialog is shown and user can change settings.
On the end, the current new data is stored in OutputDevMode where you can find it.

If the function displays the property sheet, the return value is either IDOK (1) or IDCANCEL (2), depending on which button the user selects.
If the function does not display the property sheet and is successful, the return value is IDOK (1).
If the function fails, the return value is less than zero.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation-factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd183576(v=vs.85).aspx
See also:

- 24.9.18 DocumentProperties(InputDevMode as WindowsDeviceModeMBS, byref OutputDevMode as WindowsDeviceModeMBS, Prompt as boolean = false, parent as window = nil) as Integer   853

24.9.20   GetDefaultPrinter as string

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetDefaultPrinter function retrieves the printer name of the default printer for the current user on the local computer.
Example:

msgbox WindowsPrinterMBS.GetDefaultPrinter

Notes: For more details on the GetDefaultPrinter function, check this website:
http://msdn.microsoft.com/en-us/library/dd144876(VS.85).aspx

Returns an empty string on any error.

24.9.21   GetForm(name as string) as WindowsPageFormatMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetForm function retrieves information about a specified form.
Example:

```
// query default printer
dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)

// get a page format
dim f as WindowsPageFormatMBS = w.GetForm("Letter")

// and show details
MsgBox f.DisplayName+" "+str(F.SizeWidth/10000,"0.0")+" x "+str(f.SizeHeight/10000,"0.0")
```

Notes: name: The string that specifies the name of the form. To get the names of the forms supported by the printer, call the GetPrinterFormats function.

On failure returns nil, on success the form details.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

If the caller is remote, and the Form Mode is 2, the StringType value of the returned WindowsPageFormatMBS will always be STRING_LANGPAIR.

24.9.22   GetJob(JobID as Integer) as WindowsPrinterJobMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Query job with given ID from printer.
Notes: Returns nil on any error.
PrinterName property must be set, so the function knows which printer.

24.9.23 GetPrinterFormats as WindowsPageFormatMBS()

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the page formats supported by this printer.
Notes: On failure the array is empty.

This function uses the EnumForms for which you can find more information on this website:
http://msdn.microsoft.com/en-us/library/dd162624(v=VS.85).aspx

24.9.24 GetPrinterSettings(Mode as Integer=2) as WindowsDeviceModeMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries printer settings.
Example:

dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)
dim dm as WindowsDeviceModeMBS = w.GetPrinterSettings

Notes: Mode can be 2 (general printer information), 8 (global settings) or 9 (user settings).

For more information on the GetPrinter function, check this website:
http://msdn.microsoft.com/en-us/library/dd144911(VS.85).aspx

24.9.25 GetPrinterTechnology as string

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the printer technology.
Example:

dim w as new WindowsPrinterMBS("my printer")
msgbox w.GetPrinterTechnology

Notes: Opens the printer, queries the print technology and returns that string.
For postscript printers the string should contain the word "postscript".
Returns an empty string on any error.

For details on this function, check the following website:
http://msdn.microsoft.com/en-us/library/dd144931(VS.85).aspx

24.9.26   OpenPrinter(PrinterName as string, admin as boolean = false) as WindowsPrinterMBS

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens the printer so you can operate on it.
Notes: Admin: whether you want to get permissions to administrate.
Returns nil on any error.

24.9.27   PauseJob(JobID as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pause the print job.
Notes: Returns true on success and false on failure.

24.9.28   PausePrinter as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pauses the printer.
Example:

```
dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name, true)

if p.PausePrinter then
MsgBox "OK"
else
MsgBox "Failed: "+p.LasterrorMessage+" "+str(p.Lasterror)
end if
```

Notes: Requires application to be run by administrator and printer opened with admin option. Else you get error 5.

Returns true on success and false on failure.

24.9.29 PrinterProperties(parent as DesktopWindow) as boolean

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

Function: The PrinterProperties function displays a printer-properties property sheet for the specified printer.
Example:

```
// get default printer
dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)

// open properties dialog
if w.PrinterProperties(window1) then
MsgBox "OK"
else
MsgBox "Failed"
end if
```

Notes: Returns true on success (dialog shows) or false on error.
Parent is the parent window for the new dialog. Can be nil.
See also:

- 24.9.30 PrinterProperties(parent as window = nil) as boolean 859

24.9.30 PrinterProperties(parent as window = nil) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop only.

Function: The PrinterProperties function displays a printer-properties property sheet for the specified printer.
Example:

```
// get default printer
dim w as new WindowsPrinterMBS(WindowsPrinterMBS.GetDefaultPrinter)

// open properties dialog
if w.PrinterProperties(window1) then
MsgBox "OK"
else
MsgBox "Failed"
end if
```

Notes: Returns true on success (dialog shows) or false on error.
Parent is the parent window for the new dialog. Can be nil.
See also:

24.9.31   PurgePrinter as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Purges the printer.
Example:

dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name, true)

if p.PurgePrinter then
MsgBox "OK"
else
MsgBox "Failed: "+p.LasterrorMessage+" "+str(p.Lasterror)
end if

Notes: Requires application to be run by administrator and printer opened with admin option. Else you get error 5.
Returns true on success and false on failure.

24.9.32   ResumeJob(JobID as Integer) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resume a paused print job.
Notes: Returns true on success and false on failure.

24.9.33   ResumePrinter as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resumes the printer.
Example:

dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name, true)

if p.ResumePrinter then
MsgBox "OK"
else
MsgBox "Failed: "+p.LasterrorMessage+" "+str(p.Lasterror)
end if

Notes: Requires application to be run by administrator and printer opened with admin option. Else you get error 5.
Returns true on success and false on failure.

24.9.34   SetDefaultPrinter(PrinterName as string) as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Changes the default printer to the given one.
Example:

call WindowsPrinterMBS.SetDefaultPrinter("My Printer")

Notes: Returns a Windows error code. (0 for success)

For details on the Windows SetDefaultPrinter method, check here:
http://msdn.microsoft.com/en-us/library/dd162971(VS.85).aspx

24.9.35   SetForm(name as string, form as WindowsPageFormatMBS) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetForm function sets the form information for the specified printer.
Notes: name: The string that specifies the form name for which the form information is set.
Form: The form data. Mode property must be either 1 (older format) or 2 (newer format).

Returns true on success and false on failure.

This is a blocking or synchronous function and might not return immediately. How quickly this function returns depends on run-time factors such as network status, print server configuration, and printer driver implementation—factors that are difficult to predict when writing an application. Calling this function from a thread that manages interaction with the user interface could make the application appear to be unresponsive.

SetForm can be called multiple times for an existing WindowsPageFormatMBS, each call adding additional pairs of DisplayName and LangId values. All languages versions of the form will get the Size and Image-ableArea values of the WindowsPageFormatMBS in the most recent call to SetForm.

If the caller is remote and the form's Mode is 2, the StringType value of the WindowsPageFormatMBS cannot be STRING_MUIDLL.

24.9.36   SetJob(JobID as Integer, job as WindowsPrinterJobMBS) as boolean

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets job parameters.
Example:

```
dim w as new WindowsPrinterMBS( WindowsPrinterMBS.GetDefaultPrinter, true)
dim jobs() as WindowsPrinterJobMBS = w.AllJobs
MsgBox str(UBound(jobs)+1)+" jobs"

// pick first
dim j as WindowsPrinterJobMBS = jobs(0)
j.Document = "Hello World"

// write back values
call w.SetJob(j.JobID, j)

// error?
if w.Lasterror<>0 then
MsgBox w.LasterrorMessage
end if
```

Notes: see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd162978(v=vs.85).aspx

24.9.37   SetPrinterSettings(value as WindowsDeviceModeMBS, Mode as Integer=2) as boolean

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets printer settings.
Example:

```
// switch default printer to A5 landscape paper

dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name, false)
```

dim d as WindowsDeviceModeMBS = p.GetPrinterSettings(9)

if d = nil then
MsgBox p.LasterrorMessage
else
d.PaperSize = d.DMPAPER_A5
d.Fields = BitwiseOr(d.Fields, d.DM_PAPERSIZE)
d.Orientation = d.DMORIENT_LANDSCAPE
d.Fields = BitwiseOr(d.Fields, d.DM_ORIENTATION)

if p.SetPrinterSettings(d, 9) then
MsgBox "OK"
else
MsgBox "Failed: "+p.LasterrorMessage
end if
end if

Notes: Mode can be 2 (general printer information), 8 (global settings) or 9 (user settings).

For more information on the GetPrinter function, check this website:
http://msdn.microsoft.com/en-us/library/dd145082(v=vs.85).aspx

24.9.38   Properties

24.9.39   Handle as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal handle to the printer.
Notes: (Read and Write property)

24.9.40   Lasterror as Integer

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last windows error code.
Notes: (Read and Write property)

24.9.41 LasterrorMessage as String

Plugin Version: 12.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The text message for the last error code.
Notes: (Read and Write property)

24.9.42 PrinterName as String

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the open printer in this object.
Example:

dim name as string = WindowsPrinterMBS.GetDefaultPrinter
dim p as WindowsPrinterMBS = WindowsPrinterMBS.OpenPrinter(name)
MsgBox p.PrinterName

Notes: (Read and Write property)

Chapter 25

Process

25.1 class WindowsProcessMemoryInfoMBS

25.1.1 class WindowsProcessMemoryInfoMBS

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to get information about the memory usage of a process on Windows.
Blog Entries

- MBS Xojo Plugins, version 17.1pr2

25.1.2 Methods

25.1.3 Constructor

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor which fills the values into the properties.
Example:

dim w as new WindowsProcessMemoryInfoMBS
msgbox "This application uses currently "+str(w.WorkingSetSize)+" Bytes of memory."

Notes: Queries the current process memory information.
See also:

25.1.4   Constructor(ProcessID as Integer)

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor which fills the values into the properties.
Notes: Queries the process memory information of the process with the given ID.
See also:

25.1.5   Properties

25.1.6   PageFaultCount as Integer

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of page faults.
Notes: (Read only property)

25.1.7   PagefileUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current space allocated for the pagefile, in bytes. Those pages may or may not be in memory.
Notes: (Read only property)

25.1.8   PeakPagefileUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak space allocated for the pagefile, in bytes.
Notes: (Read only property)

25.1.9   PeakWorkingSetSize as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak working set size, in bytes.
Notes: (Read only property)

25.1.10   ProcessID as Integer

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The process ID of the process queried.
Notes: Value is 0 if the query failed.
(Read only property)

25.1.11   QuotaNonPagedPoolUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak nonpaged pool usage, in bytes.
Notes: (Read only property)

25.1.12   QuotaPagedPoolUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current paged pool usage, in bytes.
Notes: (Read only property)

25.1.13   QuotaPeakNonPagedPoolUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current nonpaged pool usage, in bytes.
Notes: (Read only property)

25.1.14   QuotaPeakPagedPoolUsage as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak paged pool usage, in bytes.
Notes: (Read only property)

25.1.15   WorkingSetSize as Int64

Plugin Version: 7.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current working set size, in bytes.
Notes: This is the memory your application uses.
(Read only property)

25.2 class WindowsProcessStatisticsMBS

25.2.1 class WindowsProcessStatisticsMBS

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to get information about process statistics on Windows.
Example:

dim c as new WindowsProcessStatisticsMBS
MsgBox str(C.PeakWorkingSetSize)

Notes: See also WindowsVMStatisticsMBS for the system memory usage.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr6

- MBS Real Studio Plugins, version 11.2pr4

- MBS Plugins 10.3 Release Notes

- MBS REALbasic Plugins, version 10.3pr7

25.2.2 Methods

25.2.3 Constructor(ProcessID as Integer = -1, Mode as Integer = 255)

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new object for given process and with given mode.
Example:

dim pid as Integer = -1
dim c as new WindowsProcessStatisticsMBS(pid, WindowsProcessStatisticsMBS.ModeMemoryCounters)
MsgBox str(C.PeakWorkingSetSize)

Notes: If ProcessID is -1, the current process is queried.
Mode can be a combination of the Mode* constants you want to query. The Mode property will later contain
the Mode* constants combination of the values which are valid.

25.2.4   Properties

25.2.5   CreationTime as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A file time that receives the creation time of the process.
Notes: Property is valid if Mode contains ModeTimes.
(Read only property)

25.2.6   CycleTime as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The sum of the cycle time of all threads of the specified process.
Notes: Property is valid if Mode contains ModeTimes.
This value can only be queried on Windows Vista or newer.
(Read only property)

25.2.7   ExitTime as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A file time that receives the exit time of the process. If the process has not exited, the content of this structure is undefined.
Notes: Property is valid if Mode contains ModeTimes.
(Read only property)

25.2.8   HandleCount as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of handles in use by the process.
Example:

```
dim w as new WindowsProcessStatisticsMBS
dim n1 as Integer = w.HandleCount

dim x as new window1 // make a new window

w = new WindowsProcessStatisticsMBS
dim n2 as Integer = w.HandleCount
```

MsgBox str(n2)+" handles in used, "+str(n2-n1)+" used by second window"

Notes: Value is -1 on an error.
(Read only property)

25.2.9   KernelTime as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A file time structure that receives the amount of time that the process has executed in kernel mode.
Notes: The time that each of the threads of the process has executed in kernel mode is determined, and then all of those times are summed together to obtain this value.
Property is valid if Mode contains ModeTimes.
(Read only property)

25.2.10   Mode as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Which values have been queried.
Notes: This value is a combination of the Mode* constants.
(Read only property)

25.2.11   OtherOperationCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of I/O operations performed, other than read and write operations.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.12   OtherTransferCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of bytes transferred during operations other than read and write operations.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.13   PageFaultCount as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of page faults.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.14   PagefileUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current space allocated for the pagefile, in bytes. Those pages may or may not be in memory.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.15   PeakPagefileUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak space allocated for the pagefile, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.16   PeakWorkingSetSize as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak working set size, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.17   ProcessID as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The process ID of the process queried.
Notes: (Read only property)

25.2.18 QuotaNonPagedPoolUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak nonpaged pool usage, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.19 QuotaPagedPoolUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current paged pool usage, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.20 QuotaPeakNonPagedPoolUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current nonpaged pool usage, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.21 QuotaPeakPagedPoolUsage as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak paged pool usage, in bytes.
Notes: Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.22 ReadOperationCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of read operations performed.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.23   ReadTransferCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of bytes read.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.24   TotalIdleTime as UInt64

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The amount of time that the system has been idle.
Notes: (Read only property)

25.2.25   TotalKernelTime as UInt64

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The amount of time that the system has spent executing in Kernel mode (including all threads in all processes, on all processors).
Notes: This time value also includes the amount of time the system has been idle. So you may want to subtract idle time to get real kernel time.
(Read only property)

25.2.26   TotalUserTime as UInt64

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The amount of time that the system has spent executing in User mode (including all threads in all processes, on all processors).
Notes: If you need to know what is the percentage of the CPU time spent for user applications, you need to read this value two times and calculate the delta.
(Read only property)

25.2.27   UserTime as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A file time structure that receives the amount of time that the process has executed in user mode.
Notes: The time that each of the threads of the process has executed in user mode is determined, and then

all of those times are summed together to obtain this value.
Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.28   WorkingSetSize as Int64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current working set size, in bytes.
Notes: This is the memory your application uses.
Property is valid if Mode contains ModeMemoryCounters.
(Read only property)

25.2.29   WriteOperationCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of write operations performed.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.30   WriteTransferCount as UInt64

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of bytes written.
Notes: Property is valid if Mode contains ModeIOCounters.
(Read only property)

25.2.31   Constants

Constants

Constant	Value	Description
ModeHandles	32	One of the mode constants.
		Whether to query the handle count.

Mode Constants

Constant	Value	Description
ModeCycleTime	8	Whether to query the cycle time.
ModeIOCounters	2	Whether to query the IOCounters.
ModeMemoryCounters	1	Whether to query the memory counters.
ModeTimes	4	Whether to query the time values.
ModeTotalTime	16	Whether to query the total CPU time values. Requires Windows XP with SP1.

25.3 class WindowsVMStatisticsMBS

25.3.1 class WindowsVMStatisticsMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Holds information about the current Windows memory status.
Notes: See also WindowsProcessStatisticsMBS for the application.
Blog Entries

- MBS REALbasic Plugins, version 10.6pr2
- MBS Plugins 10.3 Release Notes
- MBS REALbasic Plugins, version 10.3pr7

25.3.2 Methods

25.3.3 Constructor

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new statistics object with current values.
Example:

```
// show values in a listbox:
dim w as new WindowsVMStatisticsMBS

list.AddRow "Pagesize: "+format(w.Pagesize,"0")
list.AddRow "MemoryLoad: "+format(w.MemoryLoad,"0")
list.AddRow "AvailablePageFileMemory: "+format(w.AvailablePageFileMemory,"0")
list.AddRow "AvailablePhysicalMemory: "+format(w.AvailablePhysicalMemory,"0")
list.AddRow "AvailableVirtualMemory: "+format(w.AvailableVirtualMemory,"0")
list.AddRow "TotalPageFileMemory: "+format(w.TotalPageFileMemory,"0")
list.AddRow "TotalPhysicalMemory: "+format(w.TotalPhysicalMemory,"0")
list.AddRow "TotalVirtualMemory: "+format(w.TotalVirtualMemory,"0")
```

25.3.4 Properties

25.3.5 AllocationGranularity as Integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The granularity for the starting address at which virtual memory can be allocated.
Notes: Typically 64 KB.

(Read only property)

25.3.6   AvailablePageFileMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the number of bytes available in the paging file.
Notes: (Read only property)

25.3.7   AvailablePhysicalMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the number of bytes of physical memory available.
Notes: (Read only property)

25.3.8   AvailableVirtualMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the number of bytes of unreserved and uncommitted memory in the user mode portion of the virtual address space of the calling process.
Notes: (Read only property)

25.3.9   Memoryload as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies a number between 0 and 100 that gives a general idea of current memory utilization, in which 0 indicates no memory use and 100 indicates full memory use.
Notes: (Read only property)

25.3.10   Pagesize as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of one memory page in memory.
Notes: On Intel 80386 or newer CPUs, it should be 4096 Bytes.
(Read only property)

25.3.11 TotalPageFileMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the total number of bytes that can be stored in the paging file. Note that this number does not represent the actual physical size of the paging file on disk.
Notes: (Read only property)

25.3.12 TotalPhysicalMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the total number of bytes of physical memory.
Notes: (Read only property)

25.3.13 TotalVirtualMemory as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the total number of bytes that can be described in the user mode portion of the virtual address space of the calling process.
Notes: Value is 0 on any error.
(Read only property)

25.4   Globals

25.4.1   GetWindowsVMStatisticsMBS as WindowsVMStatisticsMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns information about the current memory status on Windows.
Example:

dim w as WindowsVMStatisticsMBS

w=GetWindowsVMStatisticsMBS
if w<>nil then

list.AddRow format(w.pagesize,"0")
list.AddRow format(w.memoryLoad,"0")
list.AddRow format(w.availablePageFileMemory,"0")
list.AddRow format(w.availablePhysicalMemory,"0")
list.AddRow format(w.availableVirtualMemory,"0")
list.AddRow format(w.totalPageFileMemory,"0")
list.AddRow format(w.totalPhysicalMemory,"0")
list.AddRow format(w.totalVirtualMemory,"0")

else

msgBox "No Windows memory statistics..."
quit
end if

Notes: See also WindowsProcessStatisticsMBS for the application.

Chapter 26

Shell

26.1 class WindowsProcessMBS

26.1.1 class WindowsProcessMBS

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to run processes on Windows.
Example:

```
// run command
dim w as new WindowsProcessMBS

w.CommandLine = "cmd /c dir"
w.CurrentDirectory = "C:"

if not w.run then
MsgBox w.LastErrorMessage
Return
end if

// wait
while w.Running
app.YieldToNextThread
wend

// show result
dim a as Integer = w.AvailableBytesOutput
dim r as string = w.ReadOutput(a)
MsgBox r
```

Notes: Can be used like shell, but with more windows specific options.
For interactive shell, you need to run cmd.exe yourself.
This shell is asynchronously. For synchrones mode, please write yourself a loop waiting for process to finish.

Please read Microsoft documentation for more details:
https://msdn.microsoft.com/en-us/library/windows/desktop/ms682425(v=vs.85).aspx

See also NSTask (Mac only), WindowsShellExecuteAsAdminMBS, WindowsShellExecuteMBS (Windows only), ConsoleExecuteMBS and ShellMBS (cross platform).
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.2

- MBS Xojo Plugins, version 21.2pr6

- MBS Xojo Plugins, version 20.0pr8

- MBS Xojo Plugins, version 19.5pr1

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.1

- MBS Xojo / Real Studio Plugins, version 17.1pr1

- Running command lines tools on Windows

Videos

- Presentation from London conference about MBS Plugins.

Xojo Developer Magazine

- 19.4, page 10: News

- 15.3, page 11: News

26.1.2   Methods

26.1.3   Close

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Closes the process.
Notes: Same as destructor.

26.1.4 PeekError(Length as Integer = 0) as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Peeks in stderr.

26.1.5 PeekOutput(Length as Integer = 0) as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Peeks in stdout.

26.1.6 ReadError(Length as Integer = 0) as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads data from stderr.
Notes: Please pass maximum number of bytes to read.

26.1.7 ReadOutput(Length as Integer = 0) as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads data from stdout.
Notes: Please pass maximum number of bytes to read.

26.1.8 Run as Boolean

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Starts the process.
Notes: Returns true on success or false on failure.

26.1.9 SetKillProcessWhenParentDies as boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets process to be killed automatically.
Notes: When Xojo application ends, the process you started is killed automatically.

Returns true if option is set or false on failure.

26.1.10   Terminate(ExitCode as Integer) as Boolean

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Terminates the app with the given exit code.
Notes: Returns true on success.

26.1.11   Write(Data as MemoryBlock) as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends data to stdin of the running process.
Notes: Returns number of bytes written.
See also:

26.1.12   Write(Data as String) as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends data to stdin of the running process.
Notes: Returns number of bytes written.
See also:

26.1.13   Properties

26.1.14   ApplicationName as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The path to the application.
Notes: Can be empty when application is part of command line.
(Read and Write property)

26.1.15 AvailableBytesError as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries number of available bytes on stderr.
Notes: (Read only property)

26.1.16 AvailableBytesOutput as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries number of available bytes on stdout.
Notes: (Read only property)

26.1.17 CommandLine as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The command line to run.
Example:

```
// Run PowerShell to query execution policy

Dim s As New WindowsProcessMBS
s.CommandLine = "powershell.exe -Command ""Get-ExecutionPolicy -List"""

if s.run then
DelayMBS 0.5

while s.Running
// wait
app.YieldToNextThread
DelayMBS 0.1
wend

dim a as Integer = s.AvailableBytesOutput
Dim output As String = s.ReadOutput(a)

Break // see in debugger

else
break // failed
end if
```

Notes: If applicationName is set, this should only provide parameters.
(Read and Write property)

26.1.18   CurrentDirectory as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current directory for the process.
Notes: You can set this before calling Run to specify the start directory.
If this parameter is "", the new process will have the same current drive and directory as the calling process.
(Read and Write property)

26.1.19   Domain as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The domain name.
Notes: You can set username and password to run app with different user. Domain is optional to specify network domain.
(Read and Write property)

26.1.20   Environment as Dictionary

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The environment variables to use.
Example:

```
// run command
dim w as new WindowsProcessMBS

w.CommandLine = "cmd /c echo %TEST%"

// set environment
dim env as new Dictionary
env.Value("TEST") = "Hello World"
w.Environment = env

if not w.run then
MsgBox w.LastErrorMessage
Return
end if
```

```
// wait
while w.Running
app.YieldToNextThread
wend

// show result
dim a as Integer = w.AvailableBytesOutput
dim r as string = w.ReadOutput(a)
MsgBox r
```

Notes: (Read and Write property)

26.1.21 ExitCode as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the exit code of the terminated process.
Notes: (Read only property)

26.1.22 LastError as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Last windows error code.
Notes: (Read only property)

26.1.23 LastErrorMessage as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The message text for the last error.
Notes: (Read only property)

26.1.24 Password as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The password.
Notes: You can set username and password to run app with different user. Domain is optional to specify network domain.

(Read and Write property)

26.1.25   ProcessHandle as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: For Windows the process handle.
Notes: (Read only property)

26.1.26   ProcessID as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The process ID of the process.
Notes: (Read only property)

26.1.27   Running as Boolean

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks whether process is running.
Notes: Returns true if running or false if not.

The process may not terminate if there is still output data in the pipes. So please read error/stdout regularly.
(Read only property)

26.1.28   ThreadHandle as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: For Windows the thread handle for the child process.
Notes: (Read only property)

26.1.29   ThreadID as Integer

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The thread ID of the main thread for the process.
Notes: (Read only property)

26.1.30   UserName as String

Plugin Version: 17.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The user name.
Notes: You can set username and password to run app with different user. Domain is optional to specify network domain.
(Read and Write property)

26.1.31   Events

26.1.32   DataAvailable(AvailableBytesOutput as Integer, AvailableBytesError as Integer)

Plugin Version: 17.1, Platform: Windows, Targets: .

Function: The event called when data is available for stdout or stderr.

26.1.33   Terminated(ExitCode as Integer)

Plugin Version: 17.1, Platform: Windows, Targets: .

Function: The event called when process terminated.

Chapter 27

Sparkle

27.1 class WinSparkleMBS

27.1.1 class WinSparkleMBS

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for WinSparkle library.
Notes: Please first use LoadLibrary, than set properties and finally call Initialize.
Optionally you can later call Check methods.

27.1.2 Methods

27.1.3 CheckUpdateWithoutUI

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks if an update is available.
Example:

WinSparkleMBS.CheckUpdateWithoutUI

Notes: No progress UI is shown to the user when checking. If an update is available, the usual "update available" window is shown; this function is *not* completely UI-less.
Use with caution, it usually makes more sense to use the automatic update checks on interval option or manual check with visible UI.
This function returns immediately.
This function respects "Skip this version" choice by the user.

27.1.4   CheckUpdateWithUI

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks if an update is available, showing progress UI to the user.
Example:

WinSparkleMBS.CheckUpdateWithUI

Notes: Normally, WinSparkle checks for updates on startup and only shows its UI when it finds an update. If the application disables this behavior, it can hook this function to "Check for updates..." menu item.

When called, background thread is started to check for updates. A small window is shown to let the user know the progress. If no update is found, the user is told so. If there is an update, the usual "update available" window is shown.
This function returns immediately.

Because this function is intended for manual, user-initiated checks for updates, it ignores "Skip this version" even if the user checked it previously.

27.1.5   CheckUpdateWithUIAndInstall

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks if an update is available, showing progress UI to the user and immediately installing the update if one is available.
Example:

WinSparkleMBS.CheckUpdateWithUIAndInstall

Notes: This is useful for the case when users should almost always use the newest version of your software. When called, WinSparkle will check for updates showing a progress UI to the user. If an update is found the update prompt will be skipped and the update will be installed immediately.
If your application expects to do something after checking for updates you may wish to use DidNotFindUpdate and UpdateCancelled events.

27.1.6 Cleanup

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Cleans up after WinSparkle.
Example:

WinSparkleMBS.Cleanup

Notes: Should be called by the app when it's shutting down. Cancels any pending Sparkle operations and shuts down its helper threads.

27.1.7 Initialize

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Starts WinSparkle.
Example:

WinSparkleMBS.Initialize

Notes: If WinSparkle is configured to check for updates on startup, proceeds to perform the check. You should only call this function when your app is initialized and shows its main window.

This call doesn't block and returns almost immediately. If an update is available, the respective UI is shown later from a separate thread.

27.1.8 LoadLibrary(File as folderitem) as boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Loads the WinSparkle DLL.
Example:

```
dim f as FolderItem = GetFolderItem("WinSparkle.dll")

if WinSparkleMBS.LoadLibrary(f) then
MsgBox "OK"
else
MsgBox "Failed to load DLL."
end if
```

Notes: Returns true on success.
See also:

27.1.9   LoadLibrary(Path as string) as boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Loads the WinSparkle DLL.
Example:

if WinSparkleMBS.LoadLibrary("Libs\WinSparkle.dll") then
MsgBox "OK"
else
MsgBox "Failed to load DLL."
end if

Notes: Returns true on success.
See also:

27.1.10   Properties

27.1.11   AppCastURL as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets URL for the app's appcast.
Example:

WinSparkleMBS.AppCastURL = "https://www.monkeybreadsoftware.test/test/test.xml"

Notes: Only http and https schemes are supported.

If this function isn't called by the app, the URL is obtained from Windows resource named "FeedURL" of type "APPCAST".

Always use HTTPS feeds, do not use unencrypted HTTP! This is necessary to prevent both leaking user information and preventing various MITM attacks.

See https://github.com/vslavik/winsparkle/wiki/Appcast-Feeds for more information about appcast feeds.
(Read and Write property)

27.1.12 AppName as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Application name.
Example:

WinSparkleMBS.CompanyName = "Monkeybread Software"
WinSparkleMBS.AppName = "Test App"
WinSparkleMBS.AppVersion = "1.2.3"

Notes: This is both shown to the user and used in HTTP User-Agent header.

CompanyName and AppName are used to determine the location of WinSparkle settings in registry. (HKCU\Software\<CompanyName>\<AppName>\WinSparkle is used.)

If not set, this value is read from VERSIONINFO/StringFileInfo resources.
(Read and Write property)

27.1.13 AppVersion as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Version of the app, as string (e.g. "1.2" or "1.2rc1").
Example:

WinSparkleMBS.CompanyName = "Monkeybread Software"
WinSparkleMBS.AppName = "Test App"
WinSparkleMBS.AppVersion = "1.2.3"

Notes: If not set, this value is read from VERSIONINFO/StringFileInfo resources.
(Read and Write property)

27.1.14   AutomaticCheckForUpdates as Boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether updates are checked automatically or only through a manual call.
Example:

WinSparkleMBS.AutomaticCheckForUpdates = true

Notes: If disabled, CheckUpdateWithUI must be used explicitly.

True to have updates checked automatically, false otherwise.
Defaults to false when not yet configured (as happens on first start).
(Read and Write property)

27.1.15   BuildVersion as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets application build version number.
Example:

WinSparkleMBS.BuildVersion = "123"

Notes: This is the internal version number that is not normally shown to the user. It can be used for finer granularity that official release versions, e.g. for interim builds.

If this function is called, then the provided *build* number is used for comparing versions; it is compared to the "version" attribute in the appcast and corresponds to OS X Sparkle's CFBundleVersion handling. If used, then the appcast must also contain the "shortVersionString" attribute with human-readable display version string. The version passed to AppVersion property corresponds to this and is used for display.
(Read and Write property)

27.1.16   CanShutdown as Boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether your app can shutdown now.
Example:

WinSparkleMBS.CanShutdown = false

Notes: By default true.
Set to false when you have something like an open edited document.
(Read and Write property)

27.1.17   CompanyName as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the vendor.
Example:

WinSparkleMBS.CompanyName = ”Monkeybread Software”
WinSparkleMBS.AppName = ”Test App”
WinSparkleMBS.AppVersion = ”1.2.3”

Notes: CompanyName and AppName are used to determine the location of WinSparkle settings in registry.
(HKCU\Software\<CompanyName>\<AppName>\WinSparkle is used.)

If not set, this value is read from VERSIONINFO/StringFileInfo resources.
(Read and Write property)

27.1.18   DSAPubPEM as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The public key in PEM format.
Notes: If not set, the framework will look into the app resources.
(Read and Write property)

27.1.19   Language as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets UI language from its ISO code.
Example:

WinSparkleMBS.Language = ”en-US”

Notes: Language and LanguageID properties set user interface language. One must be set before Initialize to have any effect. If none of them is called, WinSparkle detects the UI language automatically.

ISO 639 language code with an optional ISO 3116 country code, e.g. "fr", "pt-PT", "pt-BR" or "pt_BR", (Read and Write property)

27.1.20   LanguageID as Integer

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets UI language from its **Win32 LANGID** code.
Notes: Language and LanguageID properties set user interface language. One must be set before Initialize to have any effect. If none of them is called, WinSparkle detects the UI language automatically. (Read and Write property)

27.1.21   LastCheckTime as Integer

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the time for the last update check.
Notes: Default value is -1, indicating that the update check has never run.
(Read only property)

27.1.22   RegistryPath as String

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The registry path where settings will be stored.
Example:

WinSparkleMBS.RegistryPath = "Software\My App\Updates"

Notes: Normally, these are stored in "HKCU\Software\<company_name>\<app_name>\WinSparkle" but if your application needs to store the data elsewhere for some reason, using this function is an alternative.
Note that path is relative to HKCU/HKLM root and the root is not part of it. For example:
(Read and Write property)

27.1.23 UpdateCheckInterval as Integer

Plugin Version: 17.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The automatic update interval.
Example:

WinSparkleMBS.UpdateCheckInterval = 7200

Notes: The interval in seconds between checks for updates. The minimum update interval is 3600 seconds (1 hour).
(Read and Write property)

27.1.24 Events

27.1.25 DidFindUpdate

Plugin Version: 17.0, Platform: Windows, Targets: .

Function: Called when an update was found.

27.1.26 DidNotFindUpdate

Plugin Version: 17.0, Platform: Windows, Targets: .

Function: Called when an update was not found.

27.1.27 Error

Plugin Version: 17.0, Platform: Windows, Targets: .

Function: Called when an update failed due to an error.

27.1.28 ShutdownRequest

Plugin Version: 17.0, Platform: Windows, Targets: .

Function: The event to ask the host to shut down immediately after launching the installer.
Notes: Its implementation should gracefully terminate the application.

27.1.29 UpdateCancelled

Plugin Version: 17.0, Platform: Windows, Targets: .

Function: The update was cancelled.

Chapter 28

Speech

28.1 class WinSpeechMBS

28.1.1 class WinSpeechMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for speech on Windows.
Notes: You need to install the "Microsoft Reader" to get this working.

see also:
http://msdn.microsoft.com/en-us/library/ms719576(v=vs.85).aspx
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.3pr6

- MBS Real Studio Plugins, version 12.3pr2

- MonkeyBread Software Releases the MBS Plugins 8.1

28.1.2 Methods

28.1.3 close

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources of this object now without waiting for Xojo to do it for you.

28.1.4   DisplayUI(type as string, title as string, parent as DesktopWindow)

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

Function: Displays the UI from the underlying text-to-speech engine's object token.
Notes: Type specifies which UI you want.
Title is optional the title of the window.
You can optionally specify a parent window.
Lasterror is set.
See also:

28.1.5   DisplayUI(type as string, title as string, parent as window)

Plugin Version: 8.1, Platform: Windows, Targets: Desktop only.

Function: Displays the UI from the underlying text-to-speech engine's object token.
Notes: Type specifies which UI you want.
Title is optional the title of the window.
You can optionally specify a parent window.
Lasterror is set.
See also:

28.1.6   DisplayUI(type as string, title as string="")

Plugin Version: 8.1, Platform: Windows, Targets: Desktop only.

Function: Displays the UI from the underlying text-to-speech engine's object token.
Notes: Type specifies which UI you want.
Title is optional the title of the window.
You can optionally specify a parent window.
Lasterror is set.
See also:

28.1.7 IsUISupported(type as string) as boolean

Plugin Version: 8.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checks if the underlying text-to-speech engine's object token supports the requested UI.
Notes: Returns whether the specified UI is supported. True indicates the UI is supported, and false indicates the UI is not supported.
Lasterror is set.

28.1.8 ListVoices as boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Makes a list of voices which is later returned using the NextVoice function.

28.1.9 NextVoice as WinVoiceMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the next voice in the list.
Notes: Returns nil on the end of the list.

28.1.10 Pause

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Pauses speech.

28.1.11 Resume

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resumes speech.

28.1.12 Skip(sentenceCount as Integer) as Integer

Plugin Version: 8.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Causes the voice to skip forward or backward the specified number of items within the text of the current speak call.

Notes: sentenceCount is the number of sentences to skip. If negative, you skip backward.
Returns the actual number of items skipped.
Lasterror is set.

28.1.13   Speak(text as string, Purge as Boolean = false, ContainsXML as boolean = false)

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Speaks a string.
Notes: The string must be a 16bit unicode string.

Version 7.8 of the plugins convert to unicode automatically.

If Purge is true an existing speech will be stopped.

If ContainsXML is true, you can include xml commands in the text. See MSDN page:
http://msdn.microsoft.com/en-us/library/ms717077%28v=vs.85%29.aspx

28.1.14   SpeakFile(file as folderitem, unicodestring as string, AudioFormat as Integer = 0, ContainsXML as boolean = false)

Plugin Version: 4.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Speaks a string to a file.
Notes: The string must be a 16bit unicode string.
File is a valid folderitem where the WAV file is stored.
The function will not return before the speaking is complete.
Lasterror is set.

Version 7.8 of the plugins convert to unicode automatically.

For AudioFormat, please use kAudioFormat* constants. Default is 22Khz Mono which is default format for speech API.

If ContainsXML is true, you can include xml commands in the text. See MSDN page:
http://msdn.microsoft.com/en-us/library/ms717077%28v=vs.85%29.aspx

28.1.15 WaitUntilDone(msTimeout as Integer)

Plugin Version: 8.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Blocks the caller until either the voice has completed speaking or the specified time interval has elapsed.
Notes: msTimeout: Timeout period in milliseconds. INFINITE may be used to prevent this method from timing out.

INFINITE is -1.

Lasterror is set.

28.1.16 Properties

28.1.17 Handle as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the speech object.
Notes: If handle is 0 there is no Speech software installed.
(Read only property)

28.1.18 IsDone as Boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the voice has finished speaking.
Notes: Lasterror is set.
(Read only property)

28.1.19 IsSpeaking as Boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the voice is speaking.
Notes: Lasterror is set.
(Read only property)

28.1.20   Lasterror as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code reported.
Notes: You may get the Speech SDK from Microsoft for more details.
(Read only property)

28.1.21   Priority as Integer

Plugin Version: 8.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current voice priority level.
Notes: The default priority is SPVPRI_NORMAL.
Lasterror is set.

Assuming an output object which implements ISpAudio, speak requests of similar priority voices are queued, and are spoken one at a time in the order they are issued. That is, speak requests from normal priority voices are put in one queue, while speak requests from alert priority voices (with priority SPVPRI_ALERT) are put in another queue.

Alert priority voices take priority over normal voices. If one or more speak requests from alert priority voices are pending, a normal voice that is speaking will be interrupted on the next alert boundary (see ISpVoice::SetAlertBoundary). When all the queued alert priority voice speak requests have been processed, the normal voice will continue.

Voices with the SPVPRI_OVER priority speak over (mix with) all other audio in the system with no synchronization. SPVPRI_OVER priority voices only mix on Windows 2000.

If the output object does not implement ISpAudio, no serialization will occur, and all voices will be treated as if their priority is SPVPRI_OVER.
(Read and Write property)

28.1.22   Rate as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current text rendering rate adjustment.
Notes: Ranges from kMinRate to kMaxRate.
(Read and Write property)

28.1.23 SyncSpeakTimeout as Integer

Plugin Version: 8.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The timeout interval in milliseconds after which, synchronous Speak calls to this instance of the voice will timeout.
Notes: Timeouts occur when waiting for access to the output object. This means that for a normal priority voice (see Priority for more information on priorities) and an output device which implements ISpAudio, a timeout may occur while waiting to reacquire the output object after an interruption by an alert priority voice. For voices of both normal and alert priorities, a timeout may also occur while waiting to reacquire the output object after the voice has been paused and resumed (see Pause and Resume).

Wait times are not accumulated - that is, if a voice waits for n milliseconds to initially acquire the output object, and is then paused and resumed, it will again wait for up to msTimeout milliseconds to reacquire the output object, not msTimeout - n milliseconds.
(Read and Write property)

28.1.24 Voice as WinVoiceMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current used voice.
Notes: Set to nil to use the default voice.
(Read and Write property)

28.1.25 Volume as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The volume for the voice.
Notes: Ranges from kMinVolume to kMaxVolume.
(Read and Write property)

28.1.26 Constants

Constants

Constant	Value	Description
kMaxRate	10	The maximum for the rate property.
kMaxVolume	100	The maximum for the volume property.
kMinRate	-10	The minimum for the rate property.
kMinVolume	0	The minimum for the volume property.
PriorityAlert	1	One of the priority constants.
PriorityNormal	0	One of the priority constants.
PriorityOver	2	One of the priority constants.
SPDUI_AddRemoveWord	"AddRemoveWord"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_AudioProperties	"AudioProperties"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_AudioVolume	"AudioVolume"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_EngineProperties	"EngineProperties"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_MicTraining	"MicTraining"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_RecoProfileProperties	"RecoProfileProperties"	One of the constants for the User Interfaces used with IsU DisplayUI.
SPDUI_UserTraining	"UserTraining"	One of the constants for the User Interfaces used with IsU DisplayUI.

Audio Formats

Constant	Value	Description
kAudioFormat_11kHz16BitMono	10	11 kHz, 16 Bit, Mono
kAudioFormat_11kHz16BitStereo	11	11 kHz, 16 Bit, Stereo
kAudioFormat_11kHz8BitMono	8	11 kHz, 8 Bit, Mono
kAudioFormat_11kHz8BitStereo	9	11 kHz, 8 Bit, Stereo
kAudioFormat_12kHz16BitMono	14	12 kHz, 16 Bit, Mono
kAudioFormat_12kHz16BitStereo	15	12 kHz, 16 Bit, Stereo
kAudioFormat_12kHz8BitMono	12	12 kHz, 8 Bit, Mono
kAudioFormat_12kHz8BitStereo	13	12 kHz, 8 Bit, Stereo
kAudioFormat_16kHz16BitMono	18	16 kHz, 16 Bit, Mono
kAudioFormat_16kHz16BitStereo	19	16 kHz, 16 Bit, Stereo
kAudioFormat_16kHz8BitMono	16	16 kHz, 8 Bit, Mono
kAudioFormat_16kHz8BitStereo	17	16 kHz, 8 Bit, Stereo
kAudioFormat_22kHz16BitMono	22	22 kHz, 16 Bit, Mono
kAudioFormat_22kHz16BitStereo	23	22 kHz, 16 Bit, Stereo
kAudioFormat_22kHz8BitMono	20	22 kHz, 8 Bit, Mono
kAudioFormat_22kHz8BitStereo	21	22 kHz, 8 Bit, Stereo
kAudioFormat_24kHz16BitMono	26	24 kHz, 16 Bit, Mono
kAudioFormat_24kHz16BitStereo	27	24 kHz, 16 Bit, Stereo
kAudioFormat_24kHz8BitMono	24	24 kHz, 8 Bit, Mono
kAudioFormat_24kHz8BitStereo	25	24 kHz, 8 Bit, Stereo
kAudioFormat_32kHz16BitMono	30	32 kHz, 16 Bit, Mono
kAudioFormat_32kHz16BitStereo	31	32 kHz, 16 Bit, Stereo
kAudioFormat_32kHz8BitMono	28	32 kHz, 8 Bit, Mono
kAudioFormat_32kHz8BitStereo	29	32 kHz, 8 Bit, Stereo
kAudioFormat_44kHz16BitMono	34	44 kHz, 16 Bit, Mono
kAudioFormat_44kHz16BitStereo	35	44 kHz, 16 Bit, Stereo
kAudioFormat_44kHz8BitMono	32	44 kHz, 8 Bit, Mono
kAudioFormat_44kHz8BitStereo	33	44 kHz, 8 Bit, Stereo
kAudioFormat_48kHz16BitMono	38	48 kHz, 16 Bit, Mono
kAudioFormat_48kHz16BitStereo	39	48 kHz, 16 Bit, Stereo
kAudioFormat_48kHz8BitMono	36	48 kHz, 8 Bit, Mono
kAudioFormat_48kHz8BitStereo	37	48 kHz, 8 Bit, Stereo
kAudioFormat_8kHz16BitMono	6	8 kHz, 16 Bit, Mono
kAudioFormat_8kHz16BitStereo	7	8 kHz, 16 Bit, Stereo
kAudioFormat_8kHz8BitMono	4	8 KHz, 8 Bit, Mono
kAudioFormat_8kHz8BitStereo	5	8 kHz, 8 Bit, Stereo
kAudioFormat_ADPCM_11kHzMono	59	ADPCM, 11 kHz, Mono
kAudioFormat_ADPCM_11kHzStereo	60	ADPCM, 11 kHz, Stereo
kAudioFormat_ADPCM_22kHzMono	61	ADPCM, 22 kHz, Mono
kAudioFormat_ADPCM_22kHzStereo	62	ADPCM, 22 kHz, Stereo
kAudioFormat_ADPCM_44kHzMono	63	ADPCM, 44 kHz, Mono
kAudioFormat_ADPCM_44kHzStereo	64	ADPCM, 44 kHz, Stereo
kAudioFormat_ADPCM_8kHzMono	57	ADPCM, 8 kHz, Mono
kAudioFormat_ADPCM_8kHzStereo	58	ADPCM, 8 kHz, Stereo
kAudioFormat_CCITT_ALaw_11kHzMono	43	CCITT, ALaw, 11 kHz, Mono
kAudioFormat_CCITT_ALaw_11kHzStereo	44	CCITT, ALaw, 11 kHz, Stereo
kAudioFormat_CCITT_ALaw_22kHzMono	45	CCITT, ALaw, 22 kHz, Mono
kAudioFormat_CCITT_ALaw_22kHzStereo	46	CCITT, ALaw, 22 kHz, Stereo
kAudioFormat_CCITT_ALaw_44kHzMono	47	CCITT, ALaw, 44 kHz, Mono
kAudioFormat_CCITT_ALaw_44kHzStereo	48	CCITT, ALaw, 44 kHz, Stereo
kAudioFormat_CCITT_ALaw_8kHzMono	41	CCITT, ALaw, 8 kHz, Mono
kAudioFormat_CCITT_ALaw_8kHzStereo	42	CCITT, ALaw, 8 kHz, Stereo
kAudioFormat_CCITT_uLaw_11kHzMono	51	CCITT, uLaw, 11 kHz, Mono
kAudioFormat_CCITT_uLaw_11kHzStereo	52	CCITT, uLaw, 11 kHz, Stereo
kAudioFormat_CCITT_uLaw_22kHzMono	53	CCITT, uLaw, 22 kHz, Mono
kAudioFormat_CCITT_uLaw_22kHzStereo	54	CCITT, uLaw, 22 kHz, Stereo
kAudioFormat_CCITT_uLaw_44kHzMono	55	CCITT, uLaw, 44 kHz, Mono
kAudioFormat_CCITT_uLaw_44kHzStereo	56	CCITT, uLaw, 44 kHz, Stereo

28.2   class WinVoiceMBS

28.2.1   class WinVoiceMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for a voice.

28.2.2   Methods

28.2.3   Description as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the voice.
Notes: Returns a 16bit unicode string.

28.2.4   Properties

28.2.5   Handle as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the voice.
Notes: (Read only property)

28.2.6   Lasterror as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code reported.
Notes: (Read only property)

Chapter 29

System

29.1 Globals

29.1.1 GetWindowsColorProfileMBS as folderitem

Plugin Version: 8.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the path to the color profile for the main screen on Windows.
Example:

dim f as folderitem

f=getwindowsColorProfileMBS

msgBox f.NativePath

Notes: Returns nil on any error.
Blog Entries

- MBS Xojo Plugins, version 21.6pr1

29.1.2 GetWindowsDisplayColorProfileMBS(DisplayIndex as Integer) as folderitem

Plugin Version: 8.6, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the path to the color profile for the given screen on Windows.
Example:

dim f as folderitem

f=GetWindowsDisplayColorProfileMBS(1)

msgBox f.NativePath

Notes: Index is zero based.
Returns nil on any error.
See also:

29.1.3   GetWindowsDisplayColorProfileMBS(DisplayName as String) as folderitem

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the path to the color profile for the given screen on Windows.
Notes: Same as GetWindowsDisplayColorProfileMBS, but without index and taking a name.
See also:

29.1.4   ExitWindowsMBS(mode as Integer) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Shutdown a Windows PC.
Example:

dim b as boolean
b=ExitWindowsMBS(0)

Notes: Mode constants:

Returns true if successfull.

The ExitWindows function returns as soon as it has initiated the shutdown. The shutdown or logoff then proceeds asynchronously.

During a shutdown or log-off operation, applications that are shut down are allowed a specific amount of time to respond to the shutdown request. If the time expires, the system displays a dialog box that allows the user to forcibly shut down the application, to retry the shutdown, or to cancel the shutdown request. If the FORCE value is specified, the system always forces applications to close and does not display the dialog box. If the FORCEIFHUNG value is specified, the system forces hung applications to close and does not

LOGOFF	0	Shuts down all processes running in the security context of the process that called the ExitWindows function. Then it logs the user off.
SHUTDOWN	1	Shuts down the system to a point at which it is safe to turn off the power. All file buffers have been flushed to disk, and all running processes have stopped.
REBOOT	2	Shuts down the system and then restarts the system.
FORCE	4	Forces processes to terminate. When this flag is set, the system does not send the WM_QUERYENDSESSION and WM_ENDSESSION messages. This can cause the applications to lose data. Therefore, you should only use this flag in an emergency.
POWEROFF	8	Shuts down the system and turns off the power. The system must support the power-off feature.
FORCEIFHUNG	16	(Windows 2000) Forces processes to terminate if they do not respond to the WM_QUERYENDSESSION or WM_ENDSESSION message. This flag is ignored if FORCE is used.

display the dialog box.

Windows NT: To shut down or restart the system, the calling process must use the AdjustTokenPrivileges function to enable the SE_SHUTDOWN_NAME privilege.
Blog Entries

- MBS Real Studio Plugins, version 12.0pr7

29.1.5 WindowsSystemMetricsMBS(what as Integer) as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WindowsSystemMetrics function retrieves various system metrics (widths and heights of display elements) and system configuration settings.
Example:

const SM_CYSMICON = 50

MsgBox str(WindowsSystemMetricsMBS(SM_CYSMICON))

Notes: All dimensions retrieved by GetSystemMetrics are in pixels.
If the function fails, the return value is zero.

Values for the what parameter:

The SM_ARRANGE setting specifies how the system arranges minimized windows, and consists of a starting position and a direction. The starting position can be one of the following values:

The direction in which to arrange can be one of the following values:

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/ms724385(v=vs.85).aspx

29.1.6   IsWindows95MBS as boolean

Plugin Version: 5.4, Platform: Windows, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** Checks whether operation system is Windows 95/98/ME.
Example:

if IsWindows95MBS then
MsgBox "Windows 95/98/ME"
else
MsgBox "no Windows 95/98/ME"
end if

Notes: Returns true or false on Windows and alway false on other platforms.
Blog Entries

- MBS Xojo Plugins, version 20.6pr4

29.1.7   IsWindowsAdminUserMBS as boolean

Plugin Version: 5.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Checkes whether the current user is an Admin User on Windows.
Example:

if IsWindowsAdminUserMBS then
MsgBox "Is admin user."
else
MsgBox "no admin user."
end if

Notes: Returns true or false on Windows and alway false on other platforms.

29.1.8   IsWindowsNTMBS as boolean

Plugin Version: 5.4, Platform: Windows, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** Checks whether operation system is Windows NT/2000/XP.
Example:

if IsWindowsNTMBS then
MsgBox "Windows NT/2000/XP"
else
MsgBox "no Windows NT/2000/XP"
end if

Notes: Returns true or false on Windows and alway false on other platforms.
Blog Entries

- MBS Xojo Plugins, version 20.6pr4

29.1.9   WindowsGetProcessIntegrityLevelMBS as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The function gets the integrity level of the current process.
Example:

msgbox str(WindowsGetProcessIntegrityLevelMBS)

Notes: Integrity level is only available on Windows Vista and newer operating systems, thus GetProcess-IntegrityLevel throws a C++ exception if it is called on systems prior to Windows Vista.

Returns the integrity level of the current process. It is usually one of these values:

Blog Entries

- MBS REALbasic Plugins Version 10.4 release notes
- Windows Admin functions
- MBS REALbasic Plugins, version 10.4pr3

29.1.10   WindowsIsApplicationRunAsAdminMBS as boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The function checks whether the current process is run as administrator.
Example:

msgbox str(WindowsIsApplicationRunAsAdminMBS)

Notes: In other words, it dictates whether the primary access token of the process belongs to user account that is a member of the local Administrators group and it is elevated.

Returns true if the primary access token of the process belongs to user account that is a member of the local Administrators group and it is elevated. Returns false if the token does not.
Blog Entries

- MBS REALbasic Plugins Version 10.4 release notes

- Windows Admin functions

- MBS REALbasic Plugins, version 10.4pr3

29.1.11   WindowsIsProcessElevatedMBS as boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The function gets the elevation information of the current process.
Example:

msgbox str(WindowsIsProcessElevatedMBS)

Notes: It dictates whether the process is elevated or not. Token elevation is only available on Windows Vista and newer operating systems, thus IsProcessElevated returns always false if it is called on systems prior to Windows Vista. It is not appropriate to use this function to determine whether a process is run as administartor.

Returns true if the process is elevated. Returns false if it is not.
Blog Entries

- MBS REALbasic Plugins Version 10.4 release notes

- Windows Admin functions

- MBS REALbasic Plugins, version 10.4pr3

29.1.12   WindowsIsUserInAdminGroupMBS as boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The function checks whether the primary access token of the process belongs to user account that is a member of the local Administrators group, even if it currently is not elevated.
Example:

msgbox str(WindowsIsUserInAdminGroupMBS)

Notes: Returns true if the primary access token of the process belongs to user account that is a member of the local Administrators group. Returns false if the token does not.
Blog Entries

- MBS REALbasic Plugins Version 10.4 release notes

- Windows Admin functions

- MBS REALbasic Plugins, version 10.4pr3

SM_ARRANGE	56	Flags specifying how the system arranged minimized windows. For more information about minimized windows, see the following Remarks section.
SM_CLEANBOOT	67	Value that specifies how the system was started: 0 Normal boot; 1 Fail-safe boot; 2 Fail-safe with network boot; Fail-safe boot (also called SafeBoot) bypasses the user's startup files.
SM_CMETRICS	83	(Windows 2000 or newer)
SM_CMONITORS	80	Number of display monitors on the desktop. (Windows 2000 or newer)
SM_CMOUSEBUTTONS	43	Number of buttons on mouse, or zero if no mouse is installed.
SM_CXBORDER	5	Width, in pixels, of a window border.
SM_CXCURSOR	13	Width, in pixels, of a cursor. The system cannot create cursors of other sizes.
SM_CXDLGFRAME	7	Thickness, in pixels, of the sizing border around the perimeter of a window that can be resized. (or SM_CXFIXEDFRAME)
SM_CXDOUBLECLK	36	Width, in pixels, of the rectangle around the location of a first click in a double-click sequence. The second click must occur within this rectangle for the system to consider the two clicks a double-click. (The two clicks must also occur within a specified time.)
SM_CXDRAG	68	Width, in pixels, of a rectangle centered on a drag point to allow for limited movement of the mouse pointer before a drag operation begins. This allows the user to click and release the mouse button easily without unintentionally starting a drag operation.
SM_CXEDGE	45	Width, in pixels, of a 3-D border.
SM_CXFRAME	32	Thickness, in pixels, of the sizing border around the perimeter of a window that can be resized. (or SM_CXSIZEFRAME)
SM_CXFULLSCREEN	16	Width of the client area for a full-screen window on the primary display monitor.
SM_CXHSCROLL	21	Width, in pixels, of the arrow bitmap on a horizontal scroll bar.
SM_CXHTHUMB	10	Width, in pixels, of the thumb box in a horizontal scroll bar.
SM_CXICON	11	The default width, in pixels, of an icon. TheLoadIcon function can load only icons of these dimensions.
SM_CXICONSPACING	38	Width, in pixels, of a grid cell for items in large icon view. Each item fits into a rectangle of this size when arranged. These values are always greater than or equal to SM_CXICON and SM_CYICON.
SM_CXMAXIMIZED	61	Default width, in pixels, of a maximized top-level window on the primary display monitor.
SM_CXMAXTRACK	59	Default maximum width, in pixels, of a window that has a caption and sizing borders. This metric refers to the entire desktop. The user cannot drag the window frame to a size larger than these dimensions.
SM_CXMENUCHECK	71	Width, in pixels, of the default menu check-mark bitmap.
SM_CXMENUSIZE	54	Width, in pixels, of menu bar buttons, such as the child window close button used in the multiple document interface.
SM_CXMIN	28	Minimum width, in pixels, of a window.
SM_CXMINIMIZED	57	Width, in pixels, of a normal minimized window.
SM_CXMINSPACING	47	Width, in pixels, of a grid cell for minimized windows. Each minimized window fits into a rectangle this size when arranged. These values are always greater than or equal to SM_CXMINIMIZED and SM_CYMINIMIZED.
SM_CXMINTRACK	34	Minimum tracking width, in pixels, of a window. The user cannot drag the window frame to a size smaller than these dimensions. A window can override these values by processing the WM_GETMINMAXINFO message.
SM_CXSCREEN	0	Width, in pixels, of the screen of the primary display monitor.
SM_CXSIZE	30	Width, in pixels, of a button in a window's caption or title bar.
SM_CXSMICON	49	Recommended width, in pixels, of a small icon. Small icons typically appear in window captions and in small icon view.
SM_CXSMSIZE	52	Width, in pixels, of small caption buttons.
SM_CXVIRTUALSCREEN	78	Width, in pixels, of the virtual screen. The virtual screen is the bounding rectangle of all display monitors. (Windows 98/ME and 2000 or newer)
SM_CXVSCROLL	2	Width, in pixels, of a vertical scroll bar; and height, in pixels, of the arrow bitmap on a vertical scroll bar.
SM_CYBORDER	6	Height, in pixels, of a window border.
SM_CYCAPTION	4	Height, in pixels, of a normal caption area.
SM_CYCURSOR	14	Height, in pixels, of a cursor. The system cannot create cursors of other sizes.
SM_CYDLGFRAME	8	Thickness, in pixels, of the frame around the perimeter of a window that has a caption but is not sizable. (or SM_CYFIXEDFRAME)
SM_CYDOUBLECLK	37	Height, in pixels, of the rectangle around the location of a first click in a double-click sequence. The second click must occur within this rectangle for the system to consider the two clicks a double-click. (The two clicks must also occur within a specified time.)
SM_CYDRAG	69	Height, in pixels, of a rectangle centered on a drag point to allow for limited movement of the mouse pointer before a drag operation begins. This allows the user to click and release the mouse button easily without unintentionally starting a drag operation.
SM_CYEDGE	46	Height, in pixels, of a 3-D border.

ARW_BOTTOMLEFT	0x0000L	Start at the lower-left corner of the screen (default position).
ARW_BOTTOMRIGHT	0x0001L	Start at the lower-right corner of the screen. Equivalent to ARW_STAR-TRIGHT.
ARW_HIDE	0x0008L	Hide minimized windows by moving them off the visible area of the screen.
ARW_TOPLEFT	0x0002L	Start at the upper-left corner of the screen. Equivalent to ARV_STARTTOP.
ARW_TOPRIGHT	0x0003L	Start at the upper-right corner of the screen. Equivalent to ARW_STARTTOP | SRW_STARTRIGHT.

ARW_DOWN	0x0004L	Arrange vertically, top to bottom.
ARW_LEFT	0x0000L	Arrange horizontally, left to right.
ARW_RIGHT	0x0000L	Arrange horizontally, right to left.
ARW_UP	0x0004L	Arrange vertically, bottom to top.

SECURITY_MANDATORY_UNTRUSTED_RID	= &h0	Means untrusted level. It is used by processes started by the Anonymous group. Blocks most write access.
SECURITY_MANDATORY_LOW_RID	= &h1000	Means low integrity level. It is used by Protected Mode Internet Explorer. Blocks write acess to most objects (such as files and registry keys) on the system.
SECURITY_MANDATORY_MEDIUM_RID	= &h2000	Means medium integrity level. It is used by normal applications being launched while UAC is enabled.
SECURITY_MANDATORY_HIGH_RID	= &h3000	Means high integrity level. It is used by administrative applications launched through elevation when UAC is enabled, or normal applications if UAC is disabled and the user is an administrator.
SECURITY_MANDATORY_SYSTEM_RID	= &h4000	Means system integrity level. It is used by services and other system-level applications (such as Wininit, Winlogon, Smss, etc.)

Chapter 30

Tapi

30.1 class ITAddressMBS

30.1.1 class ITAddressMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for an address.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

30.1.2 Methods

30.1.3 Calls as ITCallInfoMBS()

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an array of calls currently active on the address.
Notes: Sets lasterror property.

30.1.4 Constructor

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

922   CHAPTER 30.   TAPI

30.1.5 CreateCall(DestAddress as string = "", AddressType as Integer = 1, MediaTypes as Integer = 0) as TAPICallControlMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new call.
Notes: Creates a new Call object that can be used to make an outgoing call and returns a TAPICallControlMBS object. The newly created call is in the StateIdle state and has no media or terminals selected.

See also MediaTypes and LineAddressType constants.

30.1.6 Properties

30.1.7 AddressName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the displayable name of the address.
Notes: Returns the containing a displayable address name.
Sets lasterror property.
(Read only property)

30.1.8 DialableAddress as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the text which can be used to connect to this address.
Notes: The text corresponds to the destination address string that another application would use to connect to this address, such as a phone number or an e-mail name.
You can use this address with the CreateCall method.
Sets lasterror property.
(Read only property)

30.1.9 DoNotDisturb as Boolean

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the current status of the do not disturb feature on the address.
Notes: The do not disturb feature may not be available on all addresses.
If true, the do not disturb feature has been activated. If false, the do not disturb feature is not active.
Sets lasterror property.

(Read and Write property)

30.1.10   Handle as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: This is an ITAddress* pointer.
(Read and Write property)

30.1.11   Lasterror as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Set by all the methods to indicate the last error.
Value is 0 if no error.
(Read and Write property)

30.1.12   LasterrorMessage as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: (Read and Write property)

30.1.13   MessageWaiting as Boolean

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Determines if the address has a message waiting.
Notes: Returns true if a message is waiting; false, if no message is waiting.
Sets lasterror property.
(Read and Write property)

30.1.14   ServiceProviderName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the name of the Telephony Service Provider (TSP) that supports this address.
Notes: For example, Unimdm.tsp for the Unimodem service provider or H323.tsp for the H323 service provider.
Sets lasterror property.
(Read only property)

30.1.15   State as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the current state of the address.
Notes: Sets lasterror property.
Can be StateInService or StateOutOfService.
(Read only property)

30.1.16   Constants

Line Address Types

Constant	Value	Description
LineAddressTypeDomainName	8	Address type is a domain name.
LineAddressTypeEmailName	4	Address type is an e-mail name.
LineAddressTypeIPAddress	16	Address type is an IP address.
LineAddressTypePhoneNumber	1	Address type is a standard phone number.
LineAddressTypeSDP	2	Address type is Session Description Protocol (SDP) conference.

Media Types

Constant	Value	Description
MediaTypeAudio	8	An audio media stream that is entering or leaving the computer. An entering media stream would typically be played on speakers, or sent to a handset device and a leaving stream would typically be captured through a microphone or handset device.
MediaTypeDataModem	16	A data media stream that is associated with a data modem.
MediaTypeG3Fax	32	A data media stream that is associated with a G3 protocol fax.
MediaTypeMultiTrack	&h10000	A stream is on a multitrack terminal.
MediaTypeVideo	&h8000	A video media stream that is entering or leaving the computer. An entering media stream would typically be rendered in a video window and a leaving media stream would typically be captured with a video camera.

States

Constant	Value	Description
StateInService	0	Normal state; the address can be used.
StateOutOfService	1	The address is temporarily out of service, but may go back into service at some time.

30.2   class ITCallInfoMBS

30.2.1   class ITCallInfoMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for information about a call.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

30.2.2   Methods

30.2.3   Constructor

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.
Notes: Lasterror is set.
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.

30.2.4   Properties

30.2.5   Address as ITAddressMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the address of the call.
Notes: Lasterror is set.
(Read only property)

30.2.6   CalledIDName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the called location.
Notes: Lasterror is set.
(Read only property)

30.2.7   CalledIDNumber as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of the called location.
Notes: Lasterror is set.
(Read only property)

30.2.8   CalledPartyFriendlyName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The called party friendly name.
Notes: Lasterror is set.
(Read only property)

30.2.9   CallerIDName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the caller.
Notes: Lasterror is set.
(Read only property)

30.2.10   CallerIDNumber as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of the caller.
Notes: Lasterror is set.
(Read only property)

30.2.11   CallingPartyID as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identifier of the calling party.
Notes: Lasterror is set.
(Read only property)

30.2.12   Comment as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A comment about the call provided by the application that originated the call. The call state must be StateIdle when setting the comment.
Notes: Lasterror is set.
(Read only property)

30.2.13   ConnectedIDName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the connected location.
Notes: Lasterror is set.
(Read only property)

30.2.14   ConnectedIDNumber as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of the connected location.
Notes: Lasterror is set.
(Read only property)

30.2.15   DisplayableAddress as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A displayable version of the called or calling address.
Notes: Lasterror is set.
(Read only property)

30.2.16   Handle as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: This is an ITCallInfo* pointer.
(Read and Write property)

30.2.17   Lasterror as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Set by all the methods to indicate the last error.
Value is 0 if no error.
(Read and Write property)

30.2.18   LasterrorMessage as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: (Read and Write property)

30.2.19   RedirectingIDName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the location that redirected the call.
Notes: Lasterror is set.
(Read only property)

30.2.20   RedirectingIDNumber as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of the location that redirected the call.
Notes: Lasterror is set.
(Read only property)

30.2.21   RedirectionIDName as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the location to which a call has been redirected.
Notes: Lasterror is set.
(Read only property)

30.2.22   RedirectionIDNumber as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of the location to which a call has been redirected.
Notes: Lasterror is set.
(Read only property)

30.2.23   State as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current state call.
Notes: Can be StateConnected, StateDisconnected, StateHold, StateIdle, StateInProgress, StateOffering or StateQueued.
Lasterror is set.
(Read only property)

30.2.24   Constants

States

Constant	Value	Description
StateConnected	2	Call has been connected to the remote end and communication can take place.
StateDisconnected	3	Call has been disconnected. There are several causes for disconnection. See the table of valid call state transitions below.
StateHold	5	The call is in the hold state.
StateIdle	0	The call has been created, but Connect has not been called yet. A call can never transition into the idle state. This is the initial state for both incoming and outgoing calls.
StateInProgress	1	Connect has been called, and the service provider is working on making a connection. This state is valid only on outgoing calls. This message is optional, because a service provider may have a call transition directly to the connected state.
StateOffering	4	A new call has appeared, and is being offered to an application. If the application has owner privileges on the call, it can either call Answer or Disconnect while the call is in the offering state. Current call privilege can be determined by calling Privilege.
StateQueued	6	The call is queued.

30.3   class TAPICallControlMBS

30.3.1   class TAPICallControlMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The interface used by the application to connect, answer, and perform basic telephony operations on a call object.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

30.3.2   Methods

30.3.3   Answer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: answers an incoming call.
Notes: This method can succeed only if the call state is StateOffering.
Lasterror is set.

30.3.4   BlindTransfer(DestAddress as String)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: performs a blind or single-step transfer of the specified call to the specified destination address.
Notes: The application must be the owner of the call. After a successful transfer, the call state transitions to StateDisconnected.

DestAddress: Text containing destination address for the transfer.
Lasterror is set.

30.3.5   Conference(otherCall as TAPICallControlMBS, sync as boolean)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a consultation call to the conference in which the current call is a participant.
Notes: If an associated ITCallHub object does not exist, it is created.

otherCall: The consultation call.
Sync: Indicates whether the call should be conferenced synchronously (true) or asynchronously (false). See

Connect method for additional explanation.
Lasterror is set.

30.3.6   Connect(sync as boolean)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Attempts to complete the connection of an outgoing call.
Notes: Sync: Boolean indicating whether connection is to be performed synchronously (true) or asynchronously (false).

If the call is asynchronous, the application will receive information about the call's progress through the TAPIMBS CallStateChanged event. Connect may return no error, but the actual connection may fail (and the application will be notified through the event).
If the call is synchronous, this method will not return until the call is in the connected state or fails.
Lasterror is set.

To make a call, please find the ITAddressMBS for the line, use CreateCall there. Than use TAPICallControlMBS.Connect here.

30.3.7   Constructor

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

30.3.8   Dial(DestAddress as String)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Dials the specified address.
Notes: DestAddress: Representation of address to be dialed. The format must conform to a standard dialable address.
Lasterror is set.

30.3.9   Disconnect(Mode as Integer)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Disconnects the call.

Notes: The call state will transition to StateDisconnected after the method completes successfully.
Pass DisconnectModeNoAnswer, DisconnectModeNormal or DisconnectModeReject.
Lasterror is set.

30.3.10   Finish(Mode as Integer)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Called on a consultation call to finish a conference or a transfer.
Notes: Pass FinishModeAsConference or FinishModeAsTransfer.
Lasterror is set.

30.3.11   HandoffDirect(ApplicationName as String)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Hands off the call to another application.
Notes: This indicates that the application no longer requires ownership of the call.

ApplicationName: Text containing the specific application name to hand off call to. Can be full path name
or executable name.
Lasterror is set.

30.3.12   HandoffIndirect(MediaType as Integer)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Hands off the call to another application based on the media type of the call.
Notes: If multiple applications have registered as able to handle the types involved, TAPI will hand off to
the highest-priority application, which is usually the one that registered first.
This indicates that the application no longer requires ownership of the call.

for media types, please check constants in ITAddressMBS class and this website:
http://msdn.microsoft.com/en-us/library/windows/desktop/ms734210(v=vs.85).aspx
Lasterror is set.

30.3.13   Hold(hold as boolean)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Places or removes the call from the hold.
Notes: If Hold is true and the method succeeds, the call state transitions to the StateHold state. If Hold is false, the call state transitions to StateConnected.
Lasterror is set.

30.3.14   ParkDirect(ParkAddress as String)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Parks the call at a specified address.
Notes: ParkAddress: Text containing the address where the call is to be parked.
Lasterror is set.

30.3.15   ParkIndirect as string

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Parks the call and returns the parked address.
Notes: Returns representation of the address where the call was parked.
Lasterror is set.

30.3.16   Pickup(GroupID as String)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Picks up a call alerting at the specified group identification.
Notes: GroupID: The group identifier to which the alerting station belongs.
Lasterror is set.

30.3.17   RemoveFromConference

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes the call from a conference if it is involved in one.
Notes: Lasterror is set.

30.3.18   SetQOS(MediaType as Integer, ServiceLevel as Integer)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the quality of service level for the call.
Notes: for media types, please check constants in ITAddressMBS class and this website:
http://msdn.microsoft.com/en-us/library/windows/desktop/ms734210(v=vs.85).aspx

Lasterror is set.

30.3.19 SwapHold(otherCall as TAPICallControlMBS)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Swaps the call (which is active) with the specified call on hold.
Notes: Swapping the active call with the call on consultation hold allows the application to toggle between these two calls. This is typical in call waiting.
OtherCall: Call, currently on hold, that is to be made active.
Lasterror is set.

30.3.20 Transfer(otherCall as TAPICallControlMBS, sync as boolean)

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Transfers the current call to the destination address.
Notes: otherCall: Other interface of consultation call created for the transfer.
Sync: Indicates whether the method should be completed synchronously (true) or asynchronously (false).
Lasterror is set.

30.3.21 Unpark

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the call from park.
Notes: To unpark a call, CreateCall must be called using as the destination address the current parked location of the call.
Lasterror is set.

30.3.22 Properties

30.3.23 Handle as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: This is an ITBasicCallControl* pointer.
(Read and Write property)

30.3.24   Lasterror as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Set by all the methods to indicate the last error.
Value is 0 if no error.
(Read and Write property)

30.3.25   LasterrorMessage as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: (Read and Write property)

30.3.26   Constants

Disconnect Modes

Constant	Value	Description
DisconnectModeNoAnswer	1	The call is being disconnected because it has not been answered.
		(For example, an application may set a certain amount of time for the user to answer the call. If the user does not answer, the application can call Disconnect with the NOANSWER code.)
		Lasterror is set.
DisconnectModeNormal	0	The call is being disconnected as part of the normal cycle of the call.
		Lasterror is set.
DisconnectModeReject	2	The user rejected the offered call.
		Lasterror is set.

Finish Modes

Constant	Value	Description
FinishModeAsConference	1	A call is being added to a conference call.
		Lasterror is set.
FinishModeAsTransfer	0	A call transfer is being finished.
		Lasterror is set.

QOS levels

Constant	Value	Description
QualityOfServerLevelBestEffort	2	Quality of service level desired is "best effort".
QualityOfServerLevelIfAvailable	1	Quality of service level desired if available.
QualityOfServerLevelNeeded	0	Quality of service level required.

30.4   class TAPIMBS

30.4.1   class TAPIMBS

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The central class for TAPI.
Notes: MBS Plugin implements TAPI (telephone API) on Windows to catch incoming calls, so you can display e.g. customer data for the phone number when customer calls. Also you can create new call objects to dial or query information on calls or addresses.

30.4.2   Methods

30.4.3   Addresses as ITAddressMBS()

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an array with all addresses.
Notes: Lasterror is set.

30.4.4   Available as boolean

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether TAPI is available.
Notes: Returns true on Windows and false elsewhere.

30.4.5   Constructor

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

30.4.6   Destructor

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.

30.4.7 ListenOnAllAddresses

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries list of addresses and registers event handlers to receive events.
Notes: Please call this method to get the IncomingCall event working.
Lasterror is set.

30.4.8 Properties

30.4.9 EventFilter as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The event filter.
Notes: The plugin sets this to EventCallNotification + EventCallState automatically.

See constants:
http://msdn.microsoft.com/en-us/library/windows/desktop/ms734238(v=vs.85).aspx
(Read and Write property)

30.4.10 Handle as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: This is an ITTAPI* pointer.
(Read and Write property)

30.4.11 Lasterror as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Set by all the methods to indicate the last error.
Value is 0 if no error.
(Read and Write property)

30.4.12   LasterrorMessage as String

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: (Read and Write property)

30.4.13   Events

30.4.14   CallStateChanged(CallInfo as ITCallInfoMBS)

Plugin Version: 14.4, Platform: Windows, Targets: .

Function: The call state changed.
Notes: The Call state has changed.

30.4.15   IncomingCall(CallInfo as ITCallInfoMBS, BasicCallControl as TAPI-CallControlMBS)

Plugin Version: 14.4, Platform: Windows, Targets: .

Function: A new call is incoming.

30.4.16   Constants

Event Types

Constant	Value	Description
EventCallNotification	&h4	A new communications session has appeared on the address and the TAPI DLL has created a new call object. This could be a result from an incoming session, a session handed off by another application, or a session being parked on the address.
EventCallState	&h8	The Call state has changed.

Chapter 31

User Notifications

31.1 class WinUserNotificationCenterMBS

31.1.1 class WinUserNotificationCenterMBS

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for the window notification center.
Blog Entries

- MBS Xojo Plugins, version 22.2pr3

- MBS Xojo Plugins, version 18.3pr1

Videos

- Presentation from London conference about MBS Plugins.

31.1.2 Methods

31.1.3 Available as Boolean

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this Windows version has required functions.
Notes: Returns true on Windows 8.1 and Windows 10.

31.1.4 configureAUMI(Company as String, Name as String, SurName as String, VersionInfo as String) as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Configures the AppUserModelID for Constructor.
Notes: Concats strings with dot between.
Please be aware of maximum ID length of 128 characters.

31.1.5 Constructor(appName as string, aumi as string)

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Please pass your AppUserModelID and application name.

31.1.6 Destructor

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.

31.1.7 HideNotification(notification as WinUserNotificationMBS)

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Hides a notification.
Notes: Boolean return value removed for MBS Plugins 22.2.

31.1.8 Setting as Integer

Plugin Version: 18.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a value that tells you whether there is an app, user, or system block that prevents the display of a toast notification.
Notes: Returns Enabled value, if you can show notifications.

When toast notifications have been disabled at more than one level, this property value reflects the setting with the largest scope. Precedence is as follows, from largest scope to smallest:

DisabledByManifest
DisabledByGroupPolicy
DisabledForUser
DisabledForApplication

If toast notifications have been disabled both by the user and by group policy, this property returns DisabledByGroupPolicy.

31.1.9   ShowNotification(notification as WinUserNotificationMBS)

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Shows a notification.
Notes: Boolean return value removed for MBS Plugins 22.2.

31.1.10   Properties

31.1.11   appName as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The application name used.
Notes: (Read only property)

31.1.12   aumi as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The unique id.
Notes: (Read only property)

31.1.13   Handle as Integer

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object handle.
Notes: (Read only property)

31.1.14   Events

31.1.15   Activated(Notification as WinUserNotificationMBS)

Plugin Version: 17.2, Platform: Windows, Targets: .

Function: The notification was activated.

31.1.16   Dismissed(Notification as WinUserNotificationMBS, Reason as Integer)

Plugin Version: 17.2, Platform: Windows, Targets: .

Function: The notification was dismissed.
Notes: See DismissalReason* constants.

31.1.17   Failed(Notification as WinUserNotificationMBS, ErrorCode as Integer)

Plugin Version: 17.2, Platform: Windows, Targets: .

Function: Called when notification failed.

31.1.18   Constants

Dismissal Reasons

Constant	Value	Description
DismissalReasonApplicationHidden	1	The application hid the notification using hide method.
DismissalReasonTimedOut	2	The notification has expired.
DismissalReasonUserCanceled	0	The user dismissed the notification.

Limitations Modes

Constant	Value	Description
SettingDisabledByGroupPolicy	3	An administrator has disabled all notifications on this computer through group policy. The group policy setting overrides the user's setting.
SettingDisabledByManifest	4	This app has not declared itself toast capable in its package.appxmanifest file. This setting is found on the manifest's Application UI page, under the Notification section. For an app to send toast, the Toast Capable option must be set to "Yes".
SettingDisabledForApplication	1	The user has disabled notifications for this app.
SettingDisabledForUser	2	The user or administrator has disabled all notifications for this user on this computer.
SettingEnabled	0	All notifications raised by this app can be displayed.

31.2   class WinUserNotificationExceptionMBS

31.2.1   class WinUserNotificationExceptionMBS

Plugin Version: 22.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The exception class for user notifications.
Notes: Subclass of the RuntimeException class.
Blog Entries

- The Top 10 from the MBS Xojo Plugins in 2022

- News from the MBS Xojo Plugins Version 22.2

31.3 class WinUserNotificationMBS

31.3.1 class WinUserNotificationMBS

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for the notification.
Example:

dim n as new WinUserNotificationMBS

n.Image = "C:\test.png"
n.text(0) = "Hello World"
n.text(1) = "Greetings from Xojo"

Blog Entries

- The Top 10 from the MBS Xojo Plugins in 2022
- Xojo 2022r4 released
- MBS Xojo Plugins, version 22.2pr3
- MBS Xojo Plugins, version 19.3pr5
- MBS Xojo Plugins, version 18.0pr7
- MBS Xojo Plugins 17.2
- MBS Xojo Plugins, version 17.2pr1

Videos

- Presentation from London conference about MBS Plugins.

Xojo Developer Magazine

- 15.4, page 8: News

31.3.2 Methods

31.3.3 Create

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates the OS object.
Notes: Please first set text/image and then when you set other properties we create the object for you.
This method creates it automatically.

31.3.4 Properties

31.3.5 ExpiresOnReboot as Boolean

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates whether the toast notification will remain in the Notification Center after a reboot.
Notes: If true, notification will expire when the computer is rebooted.

Requires Windows 10, version 1903 (introduced in 10.0.18362.0)
(Read and Write property)

31.3.6 Group as String

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets the group identifier for the notification.
Notes: In previous versions of Windows, the group property can be a maximum length of 16 characters long. However, in the Creators Update (15063), the maximum limit was increased to 64 characters.
(Read and Write property)

31.3.7 Handle as Integer

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

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

31.3.8 Image as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The image file path.
Notes: Leave empty if you don't need a picture.

May not be a network path as system needs to load it independent of your permissions.
Maybe better use image coming with application or put one in temp folder.
(Read and Write property)

31.3.9 NotificationMirroring as Integer

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets a value that specifies whether notification mirroring is allowed.
Notes: Requires Windows 10 Anniversary Edition (introduced in 10.0.14393.0)
(Read and Write property)

31.3.10 Priority as Integer

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets the priority of the toast notification.
Notes: Requires Windows 10 Creators Update (introduced in 10.0.15063.0)

The priority setting provides hints on how and at what urgency level a notification should be presented to the user (whether to wake up the screen, etc). Whether the notification is displayed in high priority is based on the state and power management policy of the device.
(Read and Write property)

31.3.11 RemoteId as String

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets a remote id.
Notes: A remote id for the notification that enables the system to correlate this notification with another one generated on another device.

Requires Windows 10 Anniversary Edition (introduced in 10.0.14393.0)
(Read and Write property)

31.3.12 SuppressPopup as Boolean

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets whether a toast's pop-up UI is displayed on the user's screen.
Notes: Set to true to suppress the popup message; otherwise, false. The default value is false, meaning the toast's pop-up message will be shown. Setting this property to true places the toast notification silently into the action center. This enables your app to communicate with the user without interrupting them.
(Read and Write property)

31.3.13   Tag as String

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets or sets the unique identifier of this notification within the notification Group.
Notes: This API returns an exception if the property is set to null or the empty string.
The tag can be maximum 16 characters long. However, the Creators Update (15063) extends this limit to
64 characters.
(Read and Write property)

31.3.14   Text as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The text to display.
Notes: Same as text(0).
(Read and Write property)
See also:

31.3.15   XMLUsed as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The XML template used.
Notes: For debugging.
(Read only property)

31.3.16   DataValues as Dictionary

Plugin Version: 22.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries or sets data values.
Example:

```
dim n as new WinUserNotificationMBS

// set it
Dim d As New Dictionary
d.Value("Hello") = "World"

n.DataValues = d
```

// query back
Dim dic As Dictionary = n.DataValues

MessageBox "Hello: "+dic.Lookup("Hello", "")

Notes: Allows you to store data for your app with the notfication.
(Read and Write computed property)

31.3.17   Text(Index as Integer) as String

Plugin Version: 17.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The text lines to display.
Notes: Index from 0 to 2.
(Read and Write computed property)
See also:

31.3.18   Constants

Notification Mirroring

Constant	Value	Description
NotificationMirroringAllowed	0	Notification mirroring is allowed.
NotificationMirroringDisabled	1	Notification mirroring is disabled.

Priorities

Constant	Value	Description
PriorityDefault	0	The notification should have default behavior in terms of delivery and display priority during connected standby mode.
PriorityHigh	1	The notification should be treated as high priority. For desktop PCs, this means during connected standby mode the incoming notification can turn on the screen for Surface-like devices if it doesn't have a closed lid detected.

Chapter 32

Window

32.1 class DesktopWindow

32.1.1 class DesktopWindow

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends Xojo's Window Class.
Example:

window1.HasNoTitleBarMBS = true

Notes: In Xojo 2005 and newer you need to use self. in front of the method as the propertyname alone is not accepted.

32.1.2 Methods

32.1.3 HideKeyboardMBS

Plugin Version: 23.2, Platforms: macOS, Windows, Targets: Desktop only.

Function: Hides the keyboard.
Notes: On Windows same as ShowKeyboardMBS, but may not launch the keyboard.

If you called ShowKeyboardMBS before, you can call this function to hide it.

32.1.4   SetWindowFeedbackSettingMBS(Feedback as Integer, value as Variant) as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Sets the feedback configuration for a window.
Example:

const FEEDBACK_TOUCH_CONTACTVISUALIZATION = 1
const FEEDBACK_PEN_BARRELVISUALIZATION = 2
const FEEDBACK_PEN_TAP = 3
const FEEDBACK_PEN_DOUBLETAP = 4
const FEEDBACK_PEN_PRESSANDHOLD = 5
const FEEDBACK_PEN_RIGHTTAP = 6
const FEEDBACK_TOUCH_TAP = 7
const FEEDBACK_TOUCH_DOUBLETAP = 8
const FEEDBACK_TOUCH_PRESSANDHOLD = 9
const FEEDBACK_TOUCH_RIGHTTAP = 10
const FEEDBACK_GESTURE_PRESSANDTAP = 11

dim r as Boolean = self.SetWindowFeedbackSettingMBS(FEEDBACK_TOUCH_TAP, true)

if r then
dim value as Boolean
dim b as Boolean = self.WindowFeedbackSettingMBS(FEEDBACK_TOUCH_TAP, value)
if b then
MsgBox "WindowFeedbackSettingMBS: "+str(value)
end if
end if

Notes: Returns true if successful; otherwise, returns false.

Value can be nil to reset value. Or true/false to set it.
Requires Windows 8 or Windows Server 2012 in desktop apps only.

Constant Name	Value	Description
FEEDBACK_TOUCH_CONTACTVISUALIZATION	1	Feedback for a touch contact event.
FEEDBACK_PEN_BARRELVISUALIZATION	2	Feedback for a pen barrel-button event.
FEEDBACK_PEN_TAP	3	Feedback for a pen tap event.
FEEDBACK_PEN_DOUBLETAP	4	Feedback for a pen double-tap event.
FEEDBACK_PEN_PRESSANDHOLD	5	Feedback for a pen press-and-hold event.
FEEDBACK_PEN_RIGHTTAP	6	Feedback for a pen right-tap event.
FEEDBACK_TOUCH_TAP	7	Feedback for a touch tap event.
FEEDBACK_TOUCH_DOUBLETAP	8	Feedback for a touch double-tap event.
FEEDBACK_TOUCH_PRESSANDHOLD	9	Feedback for a touch press-and-hold event.
FEEDBACK_TOUCH_RIGHTTAP	10	Feedback for a touch right-tap event.
FEEDBACK_GESTURE_PRESSANDTAP	11	Feedback for a press-and-tap gesture.

32.1.5 SetWindowIconMBS(Type as integer, File as FolderItem, IconID as integer) as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Sets the window icon using the icon in the file.
Example:

call window1.SetWindowIconMBS(0,getfolderitem("icon.ico"),1)

Notes: Returns true on success and false on failure.
What image depths are used and supported depends on the Windows version.

Type is 0 for a small icon and 1 for a big icon.
See also:

- 32.1.6 SetWindowIconMBS(Type as integer, Icon as Picture, Mask as Picture) as Boolean   955

32.1.6 SetWindowIconMBS(Type as integer, Icon as Picture, Mask as Picture) as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Sets the window icon using the given picture with mask.
Example:

dim p as Picture
dim m as Picture

// random colored circle image
p=New Picture(16,16,32)
p.Graphics.ForeColor=&c000000
p.Graphics.FillRect 0,0,16,16
p.Graphics.ForeColor=rgb(rnd*256,rnd*256,rnd*256)
p.Graphics.Filloval 0,0,16,16

// circle mask
m=New Picture(16,16,32)
m.Graphics.ForeColor=&cFFFFFF // transparent
m.Graphics.Fillrect 0,0,16,16
m.Graphics.ForeColor=&c000000 // color
m.Graphics.Filloval 0,0,16,16

Canvas1.Backdrop=p // show in canvas
Canvas2.Backdrop=m

call window1.SetWindowIconMBS(0,p,m)

Notes: The mask picture ist converted to black/white.
What image depths are used and supported depends on the Windows version.
Returns true on success and false on failure.

Possible values for type:

ICON_BIG = 1	Set the large icon for the window.
ICON_SMALL = 0	Set the small icon for the window.

See also:

32.1.7   SetWindowMaskMBS(p as picture, redraw as Boolean, transparentColor as color) as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Sets the mask of a window.
Notes: Redraw decides whether the window is redrawn after it has been changed.
Returns true if successfull.

If you want to do the same on Mac OS X, check the example projects "transparent window" and the Photoshop Splash Screen example.

32.1.8   ShowKeyboardMBS

Plugin Version: 23.2, Platforms: macOS, Windows, Targets: Desktop only.

Function: Shows the keyboard viewer.
Notes: On Windows you can trigger this a second time to hide it.

See also ShowCharacterPaletteMBS function.

32.1.9 WinAnimateWindowMBS(Flags as integer, Time as integer=200) as boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: The AnimateWindow function enables you to produce special effects when showing or hiding windows.
Example:

dim flags as Integer

const AW_SLIDE = &h040000 // Uses slide animation. By default, roll animation is used. This flag is ignored when used with AW_CENTER.
const AW_ACTIVATE = &h020000 // Activates the window. Do not use this value with AW_HIDE.
const AW_BLEND = &h080000 // Uses a fade effect. This flag can be used only if hwnd is a top-level window.
const AW_HIDE = &h010000 //Hides the window. By default, the window is shown.
const AW_CENTER = &h10 // Makes the window appear to collapse inward if AW_HIDE is used or expand outward if the AW_HIDE is not used. The various direction flags have no effect.
const AW_HOR_POSITIVE = 1 // Animates the window from left to right. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_HOR_NEGATIVE = 2 // Animates the window from right to left. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_VER_POSITIVE = 4 // Animates the window from top to bottom. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_VER_NEGATIVE = 8 // Animates the window from bottom to top. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.

window2.hide

flags=BitwiseOr(AW_SLIDE, AW_HOR_POSITIVE)

if window2.WinAnimateWindowMBS(flags, 1000) then
window2.show
else
MsgBox "Animatin failed."
end if

Notes: There are four types of animation: roll, slide, collapse or expand, and alpha-blended fade.

self: The window to animate. The calling thread must own this window.
Time: Specifies how long it takes to play the animation, in milliseconds. Typically, an animation takes 200 milliseconds to play.
Flags: Specifies the type of animation. This parameter can be one or more of the following values. Note that, by default, these flags take effect when showing a window. To take effect when hiding a window, use AW_HIDE and a bitwiseor operator with the appropriate flags.

AW_SLIDE	= &h040000	Uses slide animation. By default, roll animation is used. This flag is ignored when used with AW_CENTER.
AW_ACTIVATE	= &h020000	Activates the window. Do not use this value with AW_HIDE.
AW_BLEND	= &h080000	Uses a fade effect. This flag can be used only if hwnd is a top-level window.
AW_HIDE	= &h010000	Hides the window. By default, the window is shown.
AW_CENTER	= &h10	Makes the window appear to collapse inward if AW_HIDE is used or expand outward if the AW_HIDE is not used. The various direction flags have no effect.
AW_HOR_POSITIVE	= 1	Animates the window from left to right. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_HOR_NEGATIVE	= 2	Animates the window from right to left. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_VER_POSITIVE	= 4	Animates the window from top to bottom. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_VER_NEGATIVE	= 8	Animates the window from bottom to top. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.

Return Value:
If the function succeeds, the return value is true.
If the function fails, the return value is false.
The function will fail in the following situations:

- If the window uses the window region. Windows XP: This does not cause the function to fail.

- If the window is already visible and you are trying to show the window.

- If the window is already hidden and you are trying to hide the window.

- If there is no direction specified for the slide or roll animation.

- When trying to animate a child window with AW_BLEND.

- If the thread does not own the window. Note that, in this case, AnimateWindow fails but GetLastError returns ERROR_SUCCESS. To get extended error information, call the GetLastError function.

To show or hide a window without special effects, use Show.
When using slide or roll animation, you must specify the direction. It can be either AW_HOR_POSITIVE, AW_HOR_NEGATIVE, AW_VER_POSITIVE, or AW_VER_NEGATIVE.
You can combine AW_HOR_POSITIVE or AW_HOR_NEGATIVE with AW_VER_POSITIVE or AW_VER_NEGATIVE to animate a window diagonally.
The window procedures for the window and its child windows should handle any WM_PRINT or WM_PRINTCLIENT messages. Dialog boxes, controls, and common controls already handle WM_PRINTCLIENT. The default window procedure already handles WM_PRINT.
If a child window is displayed partially clipped, when it is animated it will have holes where it is clipped.
AnimateWindow supports RTL windows.
Avoid animating a window that has a drop shadow because it produces visually distracting, jerky animations.

32.1.10 WindowFeedbackSettingMBS(Feedback as Integer, byref value as boolean, IncludeAncestors as Boolean = false) as Boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Retrieves the feedback configuration for a window.
Notes: IncludeAncestors: Specifies that the parent window chain should be checked.

Returns true if the specified feedback setting is configured on the specified window. Otherwise, it returns false (and config won't be modified).

Requires Windows 8 [desktop apps only] or Windows Server 2012 [desktop apps only] .

Constant Name	Value	Description
FEEDBACK_TOUCH_CONTACTVISUALIZATION	1	Feedback for a touch contact event.
FEEDBACK_PEN_BARRELVISUALIZATION	2	Feedback for a pen barrel-button event.
FEEDBACK_PEN_TAP	3	Feedback for a pen tap event.
FEEDBACK_PEN_DOUBLETAP	4	Feedback for a pen double-tap event.
FEEDBACK_PEN_PRESSANDHOLD	5	Feedback for a pen press-and-hold event.
FEEDBACK_PEN_RIGHTTAP	6	Feedback for a pen right-tap event.
FEEDBACK_TOUCH_TAP	7	Feedback for a touch tap event.
FEEDBACK_TOUCH_DOUBLETAP	8	Feedback for a touch double-tap event.
FEEDBACK_TOUCH_PRESSANDHOLD	9	Feedback for a touch press-and-hold event.
FEEDBACK_TOUCH_RIGHTTAP	10	Feedback for a touch right-tap event.
FEEDBACK_GESTURE_PRESSANDTAP	11	Feedback for a press-and-tap gesture.

32.1.11 WinHideTooltipMBS as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Hides all tooltips.
Notes: You can call that in window.deactivate event to hide tooltips which did not hide themselves.
Returns number of hidden tooltips.

32.1.12 Properties

32.1.13 WinTopMostWindowMBS as boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop only.

Function: Whether a window is staying on the top of the other windows.
Example:

window1.WinTopMostWindowMBS=true

Notes: If true the window stays in front of other windows. Default is false for Xojo windows and true for overlays.
(Read and Write computed property)

32.2 class Window

32.2.1 class Window

Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends Xojo's Window Class.
Example:

window1.HasNoTitleBarMBS = true

Notes: In Xojo 2005 and newer you need to use self. in front of the method as the propertyname alone is not accepted.

32.2.2 Methods

32.2.3 HideKeyboardMBS

Plugin Version: 23.2, Platforms: macOS, Windows, Targets: Desktop only.

Function: Hides the keyboard.
Notes: On Windows same as ShowKeyboardMBS, but may not launch the keyboard.

If you called ShowKeyboardMBS before, you can call this function to hide it.
Blog Entries

- News from the MBS Xojo Plugins Version 23.2

- MBS Xojo Plugins, version 23.2pr1

32.2.4 SetWindowFeedbackSettingMBS(Feedback as Integer, value as Variant) as Boolean

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Sets the feedback configuration for a window.
Example:

const FEEDBACK_TOUCH_CONTACTVISUALIZATION = 1
const FEEDBACK_PEN_BARRELVISUALIZATION = 2
const FEEDBACK_PEN_TAP = 3
const FEEDBACK_PEN_DOUBLETAP = 4
const FEEDBACK_PEN_PRESSANDHOLD = 5
const FEEDBACK_PEN_RIGHTTAP = 6

const FEEDBACK_TOUCH_TAP = 7
const FEEDBACK_TOUCH_DOUBLETAP = 8
const FEEDBACK_TOUCH_PRESSANDHOLD = 9
const FEEDBACK_TOUCH_RIGHTTAP = 10
const FEEDBACK_GESTURE_PRESSANDTAP = 11

dim r as Boolean = self.SetWindowFeedbackSettingMBS(FEEDBACK_TOUCH_TAP, true)

if r then
dim value as Boolean
dim b as Boolean = self.WindowFeedbackSettingMBS(FEEDBACK_TOUCH_TAP, value)
if b then
MsgBox "WindowFeedbackSettingMBS: "+str(value)
end if
end if

Notes: Returns true if successful; otherwise, returns false.

Value can be nil to reset value. Or true/false to set it.
Requires Windows 8 or Windows Server 2012 in desktop apps only.

Constant Name	Value	Description
FEEDBACK_TOUCH_CONTACTVISUALIZATION	1	Feedback for a touch contact event.
FEEDBACK_PEN_BARRELVISUALIZATION	2	Feedback for a pen barrel-button event.
FEEDBACK_PEN_TAP	3	Feedback for a pen tap event.
FEEDBACK_PEN_DOUBLETAP	4	Feedback for a pen double-tap event.
FEEDBACK_PEN_PRESSANDHOLD	5	Feedback for a pen press-and-hold event.
FEEDBACK_PEN_RIGHTTAP	6	Feedback for a pen right-tap event.
FEEDBACK_TOUCH_TAP	7	Feedback for a touch tap event.
FEEDBACK_TOUCH_DOUBLETAP	8	Feedback for a touch double-tap event.
FEEDBACK_TOUCH_PRESSANDHOLD	9	Feedback for a touch press-and-hold event.
FEEDBACK_TOUCH_RIGHTTAP	10	Feedback for a touch right-tap event.
FEEDBACK_GESTURE_PRESSANDTAP	11	Feedback for a press-and-tap gesture.

Blog Entries

- MBS Xojo Plugins, version 17.5pr6

Videos

- Presentation from London conference about MBS Plugins.

32.2.5   SetWindowIconMBS(Type as Integer, File as FolderItem, IconID as Integer) as Boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop only.

Function: Sets the window icon using the icon in the file.
Example:

call window1.SetWindowIconMBS(0,getfolderitem("icon.ico"),1)

Notes: Returns true on success and false on failure.
What image depths are used and supported depends on the Windows version.

Type is 0 for a small icon and 1 for a big icon.
See also:

32.2.6 SetWindowIconMBS(Type as Integer, Icon as Picture, Mask as Picture) as Boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop only.

Function: Sets the window icon using the given picture with mask.
Example:

dim p as Picture
dim m as Picture

// random colored circle image
p=New Picture(16,16,32)
p.Graphics.ForeColor=&c000000
p.Graphics.FillRect 0,0,16,16
p.Graphics.ForeColor=rgb(rnd*256,rnd*256,rnd*256)
p.Graphics.Filloval 0,0,16,16

// circle mask
m=New Picture(16,16,32)
m.Graphics.ForeColor=&cFFFFFF // transparent
m.Graphics.Fillrect 0,0,16,16
m.Graphics.ForeColor=&c000000 // color
m.Graphics.Filloval 0,0,16,16

Canvas1.Backdrop=p // show in canvas
Canvas2.Backdrop=m

call window1.SetWindowIconMBS(0,p,m)

Notes: The mask picture ist converted to black/white.

What image depths are used and supported depends on the Windows version.
Returns true on success and false on failure.

Possible values for type:

ICON_BIG = 1	Set the large icon for the window.
ICON_SMALL = 0	Set the small icon for the window.

See also:

- 32.2.5 SetWindowIconMBS(Type as Integer, File as FolderItem, IconID as Integer) as Boolean   962

32.2.7   SetWindowMaskMBS(p as picture, redraw as Boolean, transparentColor as color) as Boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop only.

Function: Sets the mask of a window.
Notes: Redraw decides whether the window is redrawn after it has been changed.
Returns true if successfull.

If you want to do the same on Mac OS X, check the example projects "transparent window" and the Photoshop Splash Screen example.
Xojo Developer Magazine

- 2.6, page 49: IRC Bots: Easy! Part II, Part II

32.2.8   ShowKeyboardMBS

Plugin Version: 23.2, Platforms: macOS, Windows, Targets: Desktop only.

Function: Shows the keyboard viewer.
Notes: On Windows you can trigger this a second time to hide it.

See also ShowCharacterPaletteMBS function.
Blog Entries

- News from the MBS Xojo Plugins Version 23.2

- MBS Xojo Plugins, version 23.2pr1

32.2.9 WinAnimateWindowMBS(Flags as Integer, Time as Integer=200) as boolean

Plugin Version: 9.4, Platform: Windows, Targets: Desktop only.

Function: The AnimateWindow function enables you to produce special effects when showing or hiding windows.
Example:

dim flags as Integer

const AW_SLIDE = &h040000 // Uses slide animation. By default, roll animation is used. This flag is ignored when used with AW_CENTER.
const AW_ACTIVATE = &h020000 // Activates the window. Do not use this value with AW_HIDE.
const AW_BLEND = &h080000 // Uses a fade effect. This flag can be used only if hwnd is a top-level window.
const AW_HIDE = &h010000 //Hides the window. By default, the window is shown.
const AW_CENTER = &h10 // Makes the window appear to collapse inward if AW_HIDE is used or expand outward if the AW_HIDE is not used. The various direction flags have no effect.
const AW_HOR_POSITIVE = 1 // Animates the window from left to right. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_HOR_NEGATIVE = 2 // Animates the window from right to left. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_VER_POSITIVE = 4 // Animates the window from top to bottom. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
const AW_VER_NEGATIVE = 8 // Animates the window from bottom to top. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.

window2.hide

flags=BitwiseOr(AW_SLIDE, AW_HOR_POSITIVE)

if window2.WinAnimateWindowMBS(flags, 1000) then
window2.show
else
MsgBox "Animatin failed."
end if

Notes: There are four types of animation: roll, slide, collapse or expand, and alpha-blended fade.

self: The window to animate. The calling thread must own this window.
Time: Specifies how long it takes to play the animation, in milliseconds. Typically, an animation takes 200 milliseconds to play.
Flags: Specifies the type of animation. This parameter can be one or more of the following values. Note that, by default, these flags take effect when showing a window. To take effect when hiding a window, use AW_HIDE and a bitwiseor operator with the appropriate flags.

AW_SLIDE	= &h040000	Uses slide animation. By default, roll animation is used. This flag is ignored when used with AW_CENTER.
AW_ACTIVATE	= &h020000	Activates the window. Do not use this value with AW_HIDE.
AW_BLEND	= &h080000	Uses a fade effect. This flag can be used only if hwnd is a top-level window.
AW_HIDE	= &h010000	Hides the window. By default, the window is shown.
AW_CENTER	= &h10	Makes the window appear to collapse inward if AW_HIDE is used or expand outward if the AW_HIDE is not used. The various direction flags have no effect.
AW_HOR_POSITIVE	= 1	Animates the window from left to right. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_HOR_NEGATIVE	= 2	Animates the window from right to left. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_VER_POSITIVE	= 4	Animates the window from top to bottom. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.
AW_VER_NEGATIVE	= 8	Animates the window from bottom to top. This flag can be used with roll or slide animation. It is ignored when used with AW_CENTER or AW_BLEND.

Return Value:
If the function succeeds, the return value is true.
If the function fails, the return value is false.
The function will fail in the following situations:

- If the window uses the window region. Windows XP: This does not cause the function to fail.

- If the window is already visible and you are trying to show the window.

- If the window is already hidden and you are trying to hide the window.

- If there is no direction specified for the slide or roll animation.

- When trying to animate a child window with AW_BLEND.

- If the thread does not own the window. Note that, in this case, AnimateWindow fails but GetLastError returns ERROR_SUCCESS. To get extended error information, call the GetLastError function.

To show or hide a window without special effects, use Show.
When using slide or roll animation, you must specify the direction. It can be either AW_HOR_POSITIVE, AW_HOR_NEGATIVE, AW_VER_POSITIVE, or AW_VER_NEGATIVE.
You can combine AW_HOR_POSITIVE or AW_HOR_NEGATIVE with AW_VER_POSITIVE or AW_VER_NEGATIVE to animate a window diagonally.
The window procedures for the window and its child windows should handle any WM_PRINT or WM_PRINTCLIENT messages. Dialog boxes, controls, and common controls already handle WM_PRINTCLIENT. The default window procedure already handles WM_PRINT.
If a child window is displayed partially clipped, when it is animated it will have holes where it is clipped.
AnimateWindow supports RTL windows.
Avoid animating a window that has a drop shadow because it produces visually distracting, jerky animations.
Blog Entries

- MBS REALbasic plug-ins version 9.4

32.2.10 WindowFeedbackSettingMBS(Feedback as Integer, byref value as boolean, IncludeAncestors as Boolean = false) as Boolean

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Retrieves the feedback configuration for a window.
Notes: IncludeAncestors: Specifies that the parent window chain should be checked.

Returns true if the specified feedback setting is configured on the specified window. Otherwise, it returns false (and config won't be modified).

Requires Windows 8 [desktop apps only] or Windows Server 2012 [desktop apps only] .

Constant Name	Value	Description
FEEDBACK_TOUCH_CONTACTVISUALIZATION	1	Feedback for a touch contact event.
FEEDBACK_PEN_BARRELVISUALIZATION	2	Feedback for a pen barrel-button event.
FEEDBACK_PEN_TAP	3	Feedback for a pen tap event.
FEEDBACK_PEN_DOUBLETAP	4	Feedback for a pen double-tap event.
FEEDBACK_PEN_PRESSANDHOLD	5	Feedback for a pen press-and-hold event.
FEEDBACK_PEN_RIGHTTAP	6	Feedback for a pen right-tap event.
FEEDBACK_TOUCH_TAP	7	Feedback for a touch tap event.
FEEDBACK_TOUCH_DOUBLETAP	8	Feedback for a touch double-tap event.
FEEDBACK_TOUCH_PRESSANDHOLD	9	Feedback for a touch press-and-hold event.
FEEDBACK_TOUCH_RIGHTTAP	10	Feedback for a touch right-tap event.
FEEDBACK_GESTURE_PRESSANDTAP	11	Feedback for a press-and-tap gesture.

Blog Entries

- MBS Xojo Plugins, version 17.5pr6

32.2.11 WinHideTooltipMBS as Integer

Plugin Version: 16.1, Platform: Windows, Targets: Desktop only.

Function: Hides all tooltips.
Notes: You can call that in window.deactivate event to hide tooltips which did not hide themselves.
Returns number of hidden tooltips.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.1pr4

32.2.12 Properties

32.2.13 WinTopMostWindowMBS as boolean

Plugin Version: 9.6, Platform: Windows, Targets: Desktop only.

Function: Whether a window is staying on the top of the other windows.
Example:

window1.WinTopMostWindowMBS=true

Notes: If true the window stays in front of other windows. Default is false for Xojo windows and true for overlays.
(Read and Write computed property)

Chapter 33

Windows

33.1 control DesktopWinPreviewControlMBS

33.1.1 control DesktopWinPreviewControlMBS

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The preview control for Windows.
Notes: Used to show PDF documents on Windows using the built-in preview handler on Windows 10 and newer.
Can also be used to use other built-in preview handlers on Windows by setting classID.

Since the control is handled by Windows, the key and mouse events are not expected to work.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.3

- PDF Viewer controls for iOS

- PDF Viewer control for Windows

- MBS Xojo Plugins, version 23.3pr2

Xojo Developer Magazine

- 21.5, page 10: News

33.1.2   Methods

33.1.3   LoadData(data as MemoryBlock)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.
See also:

33.1.4   LoadData(data as string)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.
See also:

33.1.5   LoadFile(file as folderitem)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given file.
Notes: Lasterror is set.
If the preview handler does not support reading file, the plugin reads the file and passes as data.

If the file path is invalid, the preview will show a file not found error message.

33.1.6   Properties

33.1.7   classID as String

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The class ID for the previewer to use.
Notes: If empty, uses PDF viewer.
Available preview classes in a typical Windows 10 installation:

Class ID	Description
{ 13D3C4B8-B179-4ebb-BF62-F704173E7448 }	Windows Contact Preview Handler
{ 1531d583-8375-4d3f-b5fb-d23bbd169f22 }	Windows TXT Previewer
{ 53BEDF0B-4E5B-4183-8DC9-B844344FA104 }	Microsoft Windows MAPI Preview Handler
{ 8a7cae0e-5951-49cb-bf20-ab3fa1e44b01 }	Windows Font previewer
{ 92dbad9f-5025-49b0-9078-2d78f935e341 }	Microsoft Windows Mail Mime Preview Handler
{ a42c2ccb-67d3-46fa-abe6-7d2f3488c7a3 }	Windows RTF Previewer
{ b9815375-5d7f-4ce2-9245-c9d4da436930 }	Microsoft Windows Mail Mime Preview Handler
{ BFD468D2-D0A0-4bdc-878C-E69C2F5B435D }	Microsoft Windows Mail Html Preview Handler
{ E64164EB-1AE0-4C50-BAEF-A413C2B3A4BC }	Microsoft 3MF Shell Thumbnail and Preview Handler
{ f8b8412b-dea3-4130-b36c-5e8be73106ac }	Microsoft Windows Mail Html Preview Handler
{ 3A84F9C2-6164-485C-A7D9-4B27F8AC009E }	Microsoft PDF Previewer
{ DC6EFB56-9CFA-464D-8880-44885D7DC193 }	Adobe PDF Preview Handler for Vista

(Read and Write property)

33.1.8 Lasterror as Integer

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

33.1.9 LasterrorString as String

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The error message for the error number in the lasterror property.
Notes: (Read only property)

33.1.10 Events

33.1.11 Configure

Plugin Version: 23.3, Platform: Windows, Targets: .

Function: The event called while initializing, where you can apply properties just before plugin reads them.
Notes: e.g. set classID here in code.

33.1.12 FocusLost

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The lost focus event.
In older Xojo versions, this event is named LostFocus.

33.1.13   FocusReceived

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The got focus event.
In older Xojo versions, this event is named GotFocus.

33.1.14   MenuBarSelected

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The enable menu items event.
In older Xojo versions, this event is named EnableMenuItems.

33.2 class DirectDrawGraphicsMBS

33.2.1 class DirectDrawGraphicsMBS

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for drawing with DirectDraw 2D into a graphics context in Xojo.
Example:

```
dim p as Picture = BitmapForCaching(me.Width, me.Height)
dim g as graphics = p.Graphics

// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

// now draw a few things
dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
call d.CreateStrokeStyle 0, 0, 0, 0, 0, d.DashStyleDash

const size = 50
const offset = 10
const delta = offset + size

d.FillRectangle offset + delta * 0, offset + 0 * delta, offset + delta * 0 + size, offset + 0 * delta + size
d.DrawRectangle offset + delta * 0, offset + 1 * delta, offset + delta * 0 + size, offset + 1 * delta + size

d.FillEllipse offset + delta * 1 + size/2, offset + 0 * delta + size/2, size /2, size /2
d.DrawEllipse offset + delta * 1 + size/2, offset + 1 * delta + size/2, size /2, size /2

d.FillRoundedRectangle offset + delta * 2, offset + 0 * delta, offset + delta * 2 + size, offset + 0 * delta + size, 20, 20
d.DrawRoundedRectangle offset + delta * 2, offset + 1 * delta, offset + delta * 2 + size, offset + 1 * delta + size, 20, 20

// save to desktop
dim f as FolderItem = SpecialFolder.Desktop.Child("directdraw.jpg")
p.Save(f, Picture.Formats.JPEG, picture.QualityMaximum)

// show in window
window1.Backdrop = p
```

Blog Entries

- News from the MBS Xojo Plugins Version 20.3

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.3

- Drawing with DirectDraw in Xojo

- MBS Xojo Plugins, version 20.3pr3

33.2.2   Methods

33.2.3   ClearStrokeStyle

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Clears stroke style.
Notes: If no stroke style is defined, a default one is used.

33.2.4   Constructor(Graphics as Graphics)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
d.DrawLine 10, 10, 190,190

Backdrop = p
```

Notes: Pass the graphics in a paint event or from a picture.

Requires Xojo 2019r2 or newer.

33.2.5 CreateSolidColorBrush(c as Color) as Boolean

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new solid color brush that has the specified color and a base opacity of 1.0.
Example:

dim d as new DirectDrawGraphicsMBS(g)

// make full blue with no alpha
call d.CreateSolidColorBrush(&c0000FFFF)

Notes: Pass the red, green, blue, and alpha values of the brush's color as a Xojo color.
See also:

33.2.6 CreateSolidColorBrush(red as Single, green as Single, blue as Single, alpha as Single = 1.0) as Boolean

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new solid color brush that has the specified color and a base opacity of 1.0.
Notes: Pass the red, green, blue, and alpha values of the brush's color.
Range of values is from 0.0 to 1.0.
See also:

33.2.7 CreateStrokeStyle(startCap as Integer = 0, endCap as Integer = 0, dashCap as Integer = 0, lineJoin as Integer = 0, miterLimit as Single = 1.0, dashStyle as Integer = 0, dashOffset as Single = 0, Dashes() as Single = nil) as Boolean

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates an stroke style that describes start cap, dash pattern, and other features of a stroke.
Notes: Creates and sets the stroke style. Returns true on success and false on failure.

33.2.8 Destructor

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

startCap	The cap applied to the start of all the open figures in a stroked geometry.
endCap	The cap applied to the end of all the open figures in a stroked geometry.
dashCap	The shape at either end of each dash segment.
lineJoin	A value that describes how segments are joined. This value is ignored for a vertex if the segment flags specify that the segment should have a smooth join.
miterLimit	The limit of the thickness of the join on a mitered corner. This value is always treated as though it is greater than or equal to 1.0.
dashStyle	A value that specifies whether the stroke has a dash pattern and, if so, the dash style.
dashOffset	A value that specifies an offset in the dash sequence. A positive dash offset value shifts the dash pattern, in units of stroke width, toward the start of the stroked geometry. A negative dash offset value shifts the dash pattern, in units of stroke width, toward the end of the stroked geometry.
Dashes	An array whose elements are set to the length of each dash and space in the dash pattern. The first element sets the length of a dash, the second element sets the length of a space, the third element sets the length of a dash, and so on. The length of each dash and space in the dash pattern is the product of the element value in the array and the stroke width.

Function: The destructor.

33.2.9   DrawEllipse(x as single, y as single, radiusX as single, radiusY as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Draws the outline of the specified ellipse using the specified stroke style.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
call d.CreateStrokeStyle 0, 0, 0, 0, 0, d.DashStyleDash
d.DrawEllipse 100, 100, 90, 90

Backdrop = p
```

Notes: x/y: The position of the ellipse to draw, in device-independent pixels.
radiusX/radiusY: The radius of the ellipse to draw, in device-independent pixels.

Uses current brush, strokeWidth and strokeStyle.

33.2.10   DrawLine(x1 as single, y1 as single, x2 as single, y2 as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Draws a line between the specified points using the specified stroke style.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
d.DrawLine 10, 10, 190,190

Backdrop = p
```

Notes: x1/y1: The start point of the line, in device-independent pixels.
x2/y2: The end point of the line, in device-independent pixels.

Uses current brush, strokeWidth and strokeStyle.

33.2.11   DrawRectangle(left as single, top as single, right as single, bottom as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Draws the outline of a rectangle that has the specified dimensions and stroke style.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
```

g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
call d.CreateStrokeStyle 0, 0, 0, 0, 0, d.DashStyleDash
d.DrawRectangle 10, 10, 190, 190

Backdrop = p

Notes: Pass the dimensions of the rectangle to draw, in device-independent pixels.

Uses current brush, strokeWidth and strokeStyle.

33.2.12   DrawRoundedRectangle(left as single, top as single, right as single, bottom as single, radiusX as single, radiusY as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Draws the outline of the specified rounded rectangle using the specified stroke style.
Example:

dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)
call d.CreateStrokeStyle 0, 0, 0, 0, 0, d.DashStyleDash
d.DrawRoundedRectangle 10, 10, 190, 190, 20, 20

Backdrop = p

Notes: Pass the dimensions of the rounded rectangle to draw, in device-independent pixels.

Uses current brush, strokeWidth and strokeStyle.

33.2.13 FillEllipse(x as single, y as single, radiusX as single, radiusY as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Paints the interior of the specified ellipse.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)

d.FillEllipse 100, 100, 90, 90

Backdrop = p
```

Notes: The position and radius, in device-independent pixels, of the ellipse to paint.
Uses current brush.

33.2.14 FillRectangle(left as single, top as single, right as single, bottom as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Paints the interior of the specified rectangle.
Example:

```
dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)

d.FillRectangle 10, 10, 190, 190

Backdrop = p
```

Notes: The dimension of the rectangle to paint, in device-independent pixels.
Uses current brush.

33.2.15   FillRoundedRectangle(left as single, top as single, right as single, bottom as single, radiusX as single, radiusY as single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Paints the interior of the specified rounded rectangle.
Example:

dim p as new Picture(200,200)
dim g as Graphics = p.Graphics
// Clear image with white
g.DrawingColor = &cFFFFFF
g.FillRectangle 0, 0, g.Width, g.Height

dim d as new DirectDrawGraphicsMBS(g)

call d.CreateSolidColorBrush(&cFF0000)

d.FillRoundedRectangle 10, 10, 190, 190, 20, 20

Backdrop = p

Notes: The dimensions of the rounded rectangle to paint, in device independent pixels.
Uses current brush.

33.2.16   GetTransform(byref m11 as Single, byref m12 as Single, byref m21 as Single, byref m22 as Single, byref dx as Single, byref dy as Single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the current transform of the render target.

33.2.17   RestoreDrawingState

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Restores the current drawing state.

Notes: Our plugin manages the stack of drawing states saved.
We raise an exception if your calls to restore are not balanced with save.

33.2.18   SaveDrawingState

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Saves the current drawing state.
Notes: Our plugin manages the stack of drawing states saved, so you can restore them later.

33.2.19   SetTransform(m11 as Single, m12 as Single, m21 as Single, m22 as Single, dx as Single, dy as Single)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Applies the specified transform to the render target, replacing the existing transformation.
Notes: All subsequent drawing operations occur in the transformed space.

33.2.20   Properties

33.2.21   AntialiasMode as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The antialiasing mode of the render target.
Notes: The antialiasing mode applies to all subsequent drawing operations, excluding text and glyph drawing operations.
To specify the antialiasing mode for text and glyph operations, use the SetTextAntialiasMode method.
(Read and Write property)

33.2.22   brushHandle as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference for current brush.
Notes: Reference to ID2D1Brush object.
(Read and Write property)

33.2.23   DPIX as Single

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Return the render target's dots per inch (DPI).
Notes: The horizontal DPI of the render target.
(Read only property)

33.2.24   DPIY as Single

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Return the render target's dots per inch (DPI).
Notes: The vertical DPI of the render target.
(Read only property)

33.2.25   factoryHandle as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference for the factory.
Notes: Reference to ID2D1Factory object.
(Read and Write property)

33.2.26   Handle as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference.
Notes: Reference to ID2D1RenderTarget object.
(Read and Write property)

33.2.27   Height as Single

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the render target in device-independent pixels.
Notes: (Read only property)

33.2.28 PixelHeight as UInt32

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the render target in device pixels.
Notes: (Read only property)

33.2.29 PixelWidth as UInt32

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the render target in device pixels.
Notes: (Read only property)

33.2.30 strokeStyleHandle as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal object reference for current stroke style.
Notes: Reference to ID2D1StrokeStyle object.
(Read and Write property)

33.2.31 strokeWidth as Single

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The stroke width to use.
Notes: Default is 1.

The width of the stroke, in device-independent pixels. The value must be greater than or equal to 0.0. If this parameter isn't specified, it defaults to 1.0. The stroke is centered on the line.
(Read and Write property)

33.2.32 TextAntialiasMode as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the antialiasing mode to use for subsequent text and glyph drawing operations.
Notes: The antialiasing mode to use for subsequent text and glyph drawing operations.
(Read and Write property)

33.2.33 Width as Single

Plugin Version: 20.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the render target in device-independent pixels.
Notes: (Read only property)

33.2.34 Constants

Anti Alias Modes

Constant	Value	Description
AntiAliasModeAliased	1	Objects are aliased in most cases. Objects are antialiased only when they are drawn to a render target created by the CreateDxgiSurfaceRenderTarget method and Direct3D multisampling has been enabled on the backing DirectX Graphics Infrastructure (DXGI) surface.
AntiAliasModePerPrimitive	0	Edges are antialiased using the Direct2D per-primitive method of high-quality antialiasing.

Cap Styles

Constant	Value	Description
CapStyleFlat	0	A cap that does not extend past the last point of the line. Comparable to cap used for objects other than lines.
CapStyleRound	0	A semicircle that has a diameter equal to the line thickness.
CapStyleSquare	0	Half of a square that has a length equal to the line thickness.
CapStyleTriangle	0	An isosceles right triangle whose hypotenuse is equal in length to the thickness of the line.

Dash Styles

Constant	Value	Description
DashStyleCustom	5	The dash pattern is specified by an array of floating-point values.
DashStyleDash	1	A dash followed by a gap of equal length. The dash and the gap are each twice as long as the stroke thickness. The equivalent dash array for D2D1_DASH_STYLE_DASH is { 2, 2 } .
DashStyleDashDot	3	A dash, followed by a gap, followed by a dot, followed by another gap. The equivalent dash array for D2D1_DASH_STYLE_DASH_DOT is { 2, 2, 0, 2 } .
DashStyleDashDotDot	4	A dash, followed by a gap, followed by a dot, followed by another gap, followed by another dot, followed by another gap. The equivalent dash array for D2D1_DASH_STYLE_DASH_DOT_DOT is { 2, 2, 0, 2, 0, 2 } .
DashStyleDot	2	A dot followed by a longer gap. The equivalent dash array for D2D1_DASH_STYLE_DOT is { 0, 2 } .
DashStyleSolid	0	A solid line with no breaks.

Line Join

Constant	Value	Description
LineJoinBevel	1	Beveled vertices.
LineJoinMiter	0	Regular angular vertices.
LineJoinMiterOrBevel	3	Regular angular vertices unless the join would extend beyond the miter limit; otherwise, beveled vertices.
LineJoinRound	2	Rounded vertices.

Text Anti Alias Modes

Constant	Value	Description
TextAntiAliasModeAliased	1	Do not use antialiasing.
TextAntiAliasModeClearType	1	Use ClearType antialiasing.
TextAntiAliasModeDefault	0	Use the system default. See Remarks.
TextAntiAliasModeGrayscale	1	Use grayscale antialiasing.

33.3 class MapiFileMBS

33.3.1 class MapiFileMBS

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class contain information about a file containing a message attachment stored as a temporary file.
Example:

```
dim file as new MapiFileMBS
dim t as TextOutputStream
dim f as FolderItem

// write a temp file
f=SpecialFolder.Desktop.Child("test.txt")
t=TextOutputStream.Create(f)
t.WriteLine "Test file content"
t.Close

// create file
file=new MapiFileMBS
file.Path=f
file.Filename="test file"
file.Position=-1

// next add it to message
```

Notes: Important:
Simple MAPI is not installed by Exchange Server 2003 or later. However, Simple MAPI is supported for use with Exchange 2003. To function properly, the underlying Exchange or Microsoft Outlook®MAPI subsystem must be properly installed on the client computer.

33.3.2 Properties

33.3.3 Filename as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The attachment filename seen by the recipient, which may differ from the filename in the Path-Name member if temporary files are being used.
Notes: If the FileName member is empty, the filename from PathName is used.
(Read and Write property)

33.3.4   Flags as Integer

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The flags.
Notes: Currently not used with the plugin.
(Read and Write property)

33.3.5   Path as FolderItem

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The path of the attached file.
Notes: When this property is set the PathName property is filled with the absolute path. If you query this property, you get the PathName property as a folderitem.
So this property was made for your convenience.
(Read and Write property)

33.3.6   PathName as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The fully qualified path of the attached file.
Notes: This path should include the disk drive letter and directory name.
(Read and Write property)

33.3.7   Position as Integer

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: An integer used to indicate where in the message text to render the attachment.
Notes: Attachments replace the character found at a certain position in the message text. That is, attachments replace the character in the MapiMessage NoteText property at offset nPosition (nPosition is zero based!). A value of -1 means the attachment position is not indicated; the client application will have to provide a way for the user to access the attachment.
(Read and Write property)

33.4   class MapiMessageMBS

33.4.1   class MapiMessageMBS

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a message.
Example:

dim m as new MapiMessageMBS
dim r as new MapiRecipientMBS
dim o as new MapiRecipientMBS

o.Address="SMTP:sender@example.com"
o.Name="Me"
o.Type=o.TypeOriginator

r.Address="SMTP:receiver@example.com"
r.Name="You"
r.Type=o.TypeTo

m.AddRecipient r
m.Originator=o
m.Subject="Test Email"
m.NoteText="Hello World"
m.Flags=0

dim SendFlags as integer = m.kSendFlagDialog + m.kSendFlagLogonDialog

title=str(m.SendMail(self, SendFlags, false))

Blog Entries

- MBS Xojo Plugins, version 18.5pr6

- Problems with killing Xojo threads with plugin calls.

- MBS Xojo / Real Studio Plugins, version 15.1pr1

33.4.2   Methods

33.4.3   AddFile(file as MapiFileMBS)

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a file to the message file list.
Notes: file must be not nil.

33.4.4 AddRecipient(recipient as MapiRecipientMBS)

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a recipient to the message.
Notes: recipient must be nil.

33.4.5 IsAvailable as boolean

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether MAPI is available.
Notes: Checks whether the library was loaded and whether the MAPI Registry key exists, so this has been initialized.
(mail account may not exist)

Returns true if MAPI can be used.

33.4.6 IsUnicodeAvailable as boolean

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether MAPI is available with unicode.
Notes: Checks whether the library was loaded and whether the MAPI Registry key exists, so this has been initialized.
(mail account may not exist)

Returns true if MAPI can be used with unicode.

33.4.7 SendMail(parent as DesktopWindow, SendFlags as Integer, Threaded as Boolean) as integer

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

Function: Sends an email.
Notes: Parameters:

parent:

The parent window for the dialogs used. If not nil the dialogs used will be modal.

SendFlags: Send flags. Default in older plugin versions was kSendFlagDialog when DisplayDialog parameter is true and kSendFlagLogonDialog if DisplayLogonDialog parameter was true.

Error codes:

const SUCCESS_SUCCESS = 0	No error.
const MAPI_USER_ABORT = 1	The user canceled one of the dialog boxes. No message was sent.
const MAPI_E_FAILURE = 2	One or more unspecified errors occurred. No message was sent.
const MAPI_E_LOGON_FAILURE = 3	There was no default logon, and the user failed to log on successfully when the logon dialog box was displayed. No message was sent.
const MAPI_E_DISK_FULL = 4	
const MAPI_E_INSUFFICIENT_MEMORY = 5	There was insufficient memory to proceed. No message was sent.
const MAPI_E_ACCESS_DENIED = 6	
const MAPI_E_TOO_MANY_SESSIONS = 8	
const MAPI_E_TOO_MANY_FILES = 9	There were too many file attachments. No message was sent.
const MAPI_E_TOO_MANY_RECIPIENTS = 10	There were too many recipients. No message was sent.
const MAPI_E_ATTACHMENT_NOT_FOUND = 11	The specified attachment was not found. No message was sent.
const MAPI_E_ATTACHMENT_OPEN_FAILURE = 12	The specified attachment could not be opened. No message was sent.
const MAPI_E_ATTACHMENT_WRITE_FAILURE = 13	
const MAPI_E_UNKNOWN_RECIPIENT = 14	
const MAPI_E_BAD_RECIPTYPE = 15	The type of a recipient was not TypeTO, TypeCC, or TypeBCC. No message was sent.
const MAPI_E_NO_MESSAGES = 16	
const MAPI_E_INVALID_MESSAGE = 17	
const MAPI_E_TEXT_TOO_LARGE = 18	The text in the message was too large. No message was sent.
const MAPI_E_INVALID_SESSION = 19	
const MAPI_E_TYPE_NOT_SUPPORTED = 20	
const MAPI_E_AMBIGUOUS_RECIPIENT = 21	A recipient matched more than one of the recipient descriptor structures and Dialog is false. No message was sent.
const MAPI_E_MESSAGE_IN_USE = 22	
const MAPI_E_NETWORK_FAILURE = 23	
const MAPI_E_INVALID_EDITFIELDS = 24	
const MAPI_E_INVALID_RECIPS = 25	One or more recipients were invalid or did not resolve to any address.
const MAPI_E_NOT_SUPPORTED = 26	

The MAPI SendMail function sends a standard message, with or without any user interaction. The profile must be configured so that MAPISendMail can open the default service providers without requiring user interaction.

Client applications can provide a full or partial list of recipient names, subject text, file attachments, or message text. If any information is missing, MAPISendMail can prompt the user for it. If no information is missing, either the message can be sent as is or the user can be prompted to verify the information, changing values if necessary.

A successful return from MAPISendMail does not necessarily imply recipient validation. The message might not have been sent to all recipients. Depending on the transport provider, recipient validation can be a lengthy process.

A ”” value for the Subject indicates that there is no text for the subject of the message. A ”” value for the

NoteText member indicates that there is no message text. Some client applications can truncate subject lines that are too long or contain carriage returns, line feeds, or form feeds.

Each paragraph should be terminated with a CR (0x0d), an LF (0x0a), or a CRLF pair (0x0d0a). MAPISend-Mail wraps lines as appropriate. If the text exceeds system limits, the function returns the MAPI_E_TEXT_TOO_LARGE value.

The MessageType property is used only by non-IPM applications. Applications that handle IPM messages can set it to "".

The number of attachments per message can be limited in some messaging systems. If the limit is exceeded, the MAPI_E_TOO_MANY_FILES value is returned. File attachments are copied to the message before MAPISendMail returns; therefore, later changes to the files do not affect the contents of the message. The files must be closed when they are copied. Do not attempt to display attachments outside the range of the message text.

Some messaging systems can limit the number of recipients per message. If the client application passes a non-nil value indicating a number of recipients exceeding the system limit, MAPISendMail returns the MAPI_E_TOO_MANY_RECIPIENTS value. If the recipient array is empty, the Dialog parameter must be true in the call to MAPISendMail.

Note that the recipients can include either an entry identifier, the recipient's name, an address, or a name and address pair. The following table shows how MAPISendMail handles the variety of information that can be specified:

Information	Action
entry identifier	No name resolution; the name and address are ignored.
name	Name resolved using the Simple MAPI resolution rules.
address	No name resolution; address is used for both message delivery and for displaying the recipient name.
name and address	No name resolution; name used only for displaying the recipient name.

Client applications that send messages to custom recipients can avoid name resolution. Such clients should set the Address member of the recipient to the custom address.

MAPISendMail does not require an originator-type recipient to send a message.

Important Simple MAPI is not installed by Exchange Server 2003 or later. However, Simple MAPI is supported for use with Exchange 2003. To function properly, the underlying Exchange or Microsoft Outlook®MAPI subsystem must be properly installed on the client computer.
See also:

33.4.8   SendMail(parent as window, SendFlags as Integer, Threaded as Boolean) as integer

Plugin Version: 18.5, Platform: Windows, Targets: Desktop only.

Function: Sends an email.
Notes: Parameters:

parent:
The parent window for the dialogs used. If not nil the dialogs used will be modal.

SendFlags: Send flags. Default in older plugin versions was kSendFlagDialog when DisplayDialog parameter is true and kSendFlagLogonDialog if DisplayLogonDialog parameter was true.

Error codes:

const SUCCESS_SUCCESS = 0	No error.
const MAPI_USER_ABORT = 1	The user canceled one of the dialog boxes. No message was sent.
const MAPI_E_FAILURE = 2	One or more unspecified errors occurred. No message was sent.
const MAPI_E_LOGON_FAILURE = 3	There was no default logon, and the user failed to log on successfully when the logon dialog box was displayed. No message was sent.
const MAPI_E_DISK_FULL = 4	
const MAPI_E_INSUFFICIENT_MEMORY = 5	There was insufficient memory to proceed. No message was sent.
const MAPI_E_ACCESS_DENIED = 6	
const MAPI_E_TOO_MANY_SESSIONS = 8	
const MAPI_E_TOO_MANY_FILES = 9	There were too many file attachments. No message was sent.
const MAPI_E_TOO_MANY_RECIPIENTS = 10	There were too many recipients. No message was sent.
const MAPI_E_ATTACHMENT_NOT_FOUND = 11	The specified attachment was not found. No message was sent.
const MAPI_E_ATTACHMENT_OPEN_FAILURE = 12	The specified attachment could not be opened. No message was sent.
const MAPI_E_ATTACHMENT_WRITE_FAILURE = 13	
const MAPI_E_UNKNOWN_RECIPIENT = 14	
const MAPI_E_BAD_RECIPTYPE = 15	The type of a recipient was not TypeTO, TypeCC, or TypeBCC. No message was sent.
const MAPI_E_NO_MESSAGES = 16	
const MAPI_E_INVALID_MESSAGE = 17	
const MAPI_E_TEXT_TOO_LARGE = 18	The text in the message was too large. No message was sent.
const MAPI_E_INVALID_SESSION = 19	
const MAPI_E_TYPE_NOT_SUPPORTED = 20	
const MAPI_E_AMBIGUOUS_RECIPIENT = 21	A recipient matched more than one of the recipient descriptor structures and Dialog is false. No message was sent.
const MAPI_E_MESSAGE_IN_USE = 22	
const MAPI_E_NETWORK_FAILURE = 23	
const MAPI_E_INVALID_EDITFIELDS = 24	
const MAPI_E_INVALID_RECIPS = 25	One or more recipients were invalid or did not resolve to any address.
const MAPI_E_NOT_SUPPORTED = 26	

The MAPI SendMail function sends a standard message, with or without any user interaction. The profile must be configured so that MAPISendMail can open the default service providers without requiring user

interaction.

Client applications can provide a full or partial list of recipient names, subject text, file attachments, or message text. If any information is missing, MAPISendMail can prompt the user for it. If no information is missing, either the message can be sent as is or the user can be prompted to verify the information, changing values if necessary.

A successful return from MAPISendMail does not necessarily imply recipient validation. The message might not have been sent to all recipients. Depending on the transport provider, recipient validation can be a lengthy process.

A "" value for the Subject indicates that there is no text for the subject of the message. A "" value for the NoteText member indicates that there is no message text. Some client applications can truncate subject lines that are too long or contain carriage returns, line feeds, or form feeds.

Each paragraph should be terminated with a CR (0x0d), an LF (0x0a), or a CRLF pair (0x0d0a). MAPISend-Mail wraps lines as appropriate. If the text exceeds system limits, the function returns the MAPI_E_TEXT_TOO_LARGE value.

The MessageType property is used only by non-IPM applications. Applications that handle IPM messages can set it to "".

The number of attachments per message can be limited in some messaging systems. If the limit is exceeded, the MAPI_E_TOO_MANY_FILES value is returned. File attachments are copied to the message before MAPISendMail returns; therefore, later changes to the files do not affect the contents of the message. The files must be closed when they are copied. Do not attempt to display attachments outside the range of the message text.

Some messaging systems can limit the number of recipients per message. If the client application passes a non-nil value indicating a number of recipients exceeding the system limit, MAPISendMail returns the MAPI_E_TOO_MANY_RECIPIENTS value. If the recipient array is empty, the Dialog parameter must be true in the call to MAPISendMail.

Note that the recipients can include either an entry identifier, the recipient's name, an address, or a name and address pair. The following table shows how MAPISendMail handles the variety of information that can be specified:

Client applications that send messages to custom recipients can avoid name resolution. Such clients should set the Address member of the recipient to the custom address.

MAPISendMail does not require an originator-type recipient to send a message.

Information	Action
entry identifier	No name resolution; the name and address are ignored.
name	Name resolved using the Simple MAPI resolution rules.
address	No name resolution; address is used for both message delivery and for displaying the recipient name.
name and address	No name resolution; name used only for displaying the recipient name.

Important Simple MAPI is not installed by Exchange Server 2003 or later. However, Simple MAPI is supported for use with Exchange 2003. To function properly, the underlying Exchange or Microsoft Outlook®MAPI subsystem must be properly installed on the client computer.
See also:

33.4.9 SendMail(SendFlags as Integer, Threaded as Boolean) as integer

Plugin Version: 18.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sends an email.
Notes: This method is without parent window for dialogs, so you can use it in a console project.
See also:

33.4.10 Properties

33.4.11 ConversationID as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string identifying the conversation thread to which the message belongs.
Notes: Some messaging systems can ignore and not return this member.
(Read and Write property)

33.4.12 DateReceived as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string indicating the date when the message was received.
Notes: The format is YYYY/MM/DD HH:MM, using a 24-hour clock.
(Read and Write property)

33.4.13   Flags as Integer

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Bitmask of message status flags.
Notes: Can be a combination of kFlagsReceiptRequested, kFlagsSent and kFlagsUnread.
(Read and Write property)

33.4.14   MessageType as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string indicating a non-IPM type of message.
Notes: Client applications can select message types for their non-IPM messages. Clients that only support IPM messages can ignore the MessageType member when reading messages and set it to empty when sending messages.
(Read and Write property)

33.4.15   NoteText as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string containing the message text.
Notes: If this member is empty, there is no message text.
(Read and Write property)

33.4.16   Originator as MapiRecipientMBS

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The reference to the originator of this message.
Notes: (Read and Write property)

33.4.17   Subject as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The text string describing the message subject, typically limited to 256 characters or less.
Notes: If this member is empty, the user has not entered subject text.
(Read and Write property)

33.4.18   UseUnicode as Boolean

Plugin Version: 18.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to use Unicode API.
Notes: Microsoft offers both unicode and ANSI interfaces.
By default UseUnicode is true and you can set it to false to use ANSI interface if needed.
If no unicode interface is needed, we automatically use the ANSI interface.
(Read and Write property)

33.4.19   Constants

Message Flags

Constant	Value	Description
kFlagsReceiptRequested	2	A receipt notification is requested. Client applications set this flag when sending a message.
		Note: Most email applications ignore the receipt notifications.
kFlagsSent	4	The message has been sent.
kFlagsUnread	1	The message has not been read.

Send Flags

Constant	Value	Description
kSendFlagDialog	8	Display a send note UI
		A dialog box should be displayed to prompt the user for recipients and other sending options. When Dialog is false, at least one recipient must be specified.
kSendFlagLogonDialog	1	Display logon UI.
		A dialog box should be displayed to prompt the user to log on if required. When the DisplayLogonDialog is false, the client application does not display a logon dialog box and returns an error value if the user is not logged on. SendMail ignores this flag if the MessageID parameter is empty.
kSendFlagNewSession	2	Don't use shared session
		Create a new session.

33.5 class MapiRecipientMBS

33.5.1 class MapiRecipientMBS

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class contains information about a message sender or recipient.
Example:

dim o as new MapiRecipientMBS

o.Address="SMTP:sender@example.com"
o.Name="Me"
o.Type=o.TypeOriginator

// next add it to message

33.5.2 Properties

33.5.3 Address as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Optional the recipient or sender's address.
Example:

dim o as new MapiRecipientMBS
o.Address="SMTP:sender@example.com"
o.Name="Me"
o.Type=o.TypeOriginator

Notes: This address is provider-specific message delivery data. Generally, the messaging system provides such addresses for inbound messages. For outbound messages, the Address member can point to an address entered by the user for a recipient not in an address book (that is, a custom recipient). The format of an address pointed to by the Address member is [address type] [e-mail address] . Examples of valid addresses are FAX:206-555-1212 and SMTP:M@X.COM.
(Read and Write property)

33.5.4 Name as String

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the recipient.

Notes: (Read and Write property)

33.5.5   Type as Integer

Plugin Version: 8.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Contains a numeric value that indicates the type of recipient.
Example:

dim o as new MapiRecipientMBS
o.Address="SMTP:you@example.com"
o.Name="You"
o.Type=o.TypeTo

Notes: Use one of the Type constants.
(Read and Write property)

33.5.6   Constants

Constants

Constant	Value	Description
TypeBCC	3	One of the constants for the type property. Indicates a recipient of a blind copy.
TypeCC	2	One of the constants for the type property. Indicates a recipient of a message copy.
TypeOriginator	0	One of the constants for the type property. Indicates the original sender of the message.
TypeTo	1	One of the constants for the type property. Indicates a primary message recipient.

33.6 class TaskDialogButtonMBS

33.6.1 class TaskDialogButtonMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a button.

33.6.2 Properties

33.6.3 Default as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this is default button.
Notes: Only one button can be default.
(Read and Write property)

33.6.4 Enabled as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to enable this button.
Notes: Default: true.

This is a live property. Setting it while dialog is open changes the button state.
(Read and Write property)

33.6.5 ID as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The unique ID for the button.
Notes: The IDs should be unique for all buttons.
(Read and Write property)

33.6.6 Text as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The caption for the button.
Notes: (Read and Write property)

33.6.7   Visible as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this button is visible.
Notes: Default true.
(Read and Write property)

33.7 class TaskDialogMBS

33.7.1 class TaskDialogMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a dialog in Windows.
Notes: This class function creates, displays, and operates a task dialog. The task dialog contains application-defined icons, messages, title, verification check box, command links, push buttons, and radio buttons.

Requires Windows Vista or newer.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3

- MBS Xojo / Real Studio plug-ins version 16.2

- MBS Xojo / Real Studio Plugins, version 16.2pr2

- New TaskDialog Class

33.7.2 Methods

33.7.3 AppendButton(button as TaskDialogButtonMBS)

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a button.
Example:

dim td as TaskDialogMBS

dim b as new TaskDialogButtonMBS
b.Text = "Hello"
b.ID = 123

td.AppendButton b

33.7.4 AppendRadioButton(button as TaskDialogButtonMBS)

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds a radio button.
Example:

dim td as TaskDialogMBS

dim b as new TaskDialogButtonMBS
b.Text = "Hello"
b.ID = 123

td.AppendRadioButton b

33.7.5   CloseDialog

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Closes the dialog.

33.7.6   FindButtonByID(ID as Integer) as TaskDialogButtonMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Finds button by ID.
Notes: This searches buttons and radiobuttons for given ID.
It will not find common buttons.
Returns button if found or nil if not found.

33.7.7   ShowDialog as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Shows the dialog.
Example:

dim td as new TaskDialogMBS

td.CommonButtons = BitwiseOr(td.kCommonButtonOK, td.kCommonButtonCancel)
td.Content = "Hello World"

if td.ShowDialog then

if td.SelectedButton = td.kIDOK then
MsgBox "OK"
else
MsgBox "Cancel"
end if

else
MsgBox "Failed"
end if

Notes: Returns true if everything worked right.
Returns false if creation of dialog failed.

Requires Windows Vista or newer.

33.7.8 Properties

33.7.9 AllowDialogCancellation as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to allow cancel.
Notes: Indicates that the dialog should be able to be closed using Alt-F4, Escape, and the title bar's close button even if no cancel button is specified in either the CommonButtons or Buttons members.
(Read and Write property)

33.7.10 CanBeMinimized as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates that the task dialog can be minimized.
Notes: (Read and Write property)

33.7.11 CollapsedControlText as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used to label the button for expanding the expandable information.
Notes: This member is ignored when the ExpandedInformation member is empty. If this member is empty and the CollapsedControlText is specified, then the CollapsedControlText value will be used for this member as well.
(Read and Write property)

33.7.12 CommonButtons as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the push buttons displayed in the task dialog.
Example:

dim td as new TaskDialogMBS

td.CommonButtons = BitwiseOr(td.kCommonButtonOK, td.kCommonButtonCancel)
td.Content = "Hello World"

if td.ShowDialog then

if td.SelectedButton = td.kIDOK then
MsgBox "OK"
else
MsgBox "Cancel"
end if

else
MsgBox "Failed"
end if

Notes: If no common buttons are specified and no custom buttons are specified through buttons array, the task dialog will contain the OK button by default. This parameter may be a combination of flags from the following group:

kCommonButtonOK	The task dialog contains the push button: OK.
kCommonButtonYes	The task dialog contains the push button: Yes.
kCommonButtonNo	The task dialog contains the push button: No.
kCommonButtonCancel	The task dialog contains the push button: Cancel. If this button is specified, the task dialog will respond to typical cancel actions (Alt-F4 and Escape).
kCommonButtonRetry	The task dialog contains the push button: Retry.
kCommonButtonClose	The task dialog contains the push button: Close.

(Read and Write property)

33.7.13 Content as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used for the dialog's primary content.
Example:

```
dim td as new TaskDialogMBS

td.CommonButtons = BitwiseOr(td.kCommonButtonOK, td.kCommonButtonCancel)
td.Content = "Hello World"

if td.ShowDialog then

if td.SelectedButton = td.kIDOK then
MsgBox "OK"
else
MsgBox "Cancel"
end if

else
MsgBox "Failed"
end if
```

Notes: If the EnableHyperlinks flag is true, then this string may contain hyperlinks in the form: <A HREF="executablestring">Hyperlink Text</A>.
WARNING: Enabling hyperlinks when using content from an unsafe source may cause security vulnerabilities.
(Read and Write property)

33.7.14 DefaultButton as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The default button for the task dialog.
Notes: This may be any of the values specified in ID of one of the buttons, or one of the IDs corresponding to the buttons specified in the CommonButtons property.

kIDCancel	Make the Cancel button the default.
kIDNo	Make the No button the default.
kIDOk	Make the OK button the default.
kIDRetry	Make the Retry button the default.
kIDYes	Make the Yes button the default.
kIDClose	Make the Close button the default.

You can also mark buttons default with their default flag.
(Read and Write property)

33.7.15   DefaultRadioButton as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The button ID of the radio button that is selected by default.
Notes: If this value does not correspond to a button ID, the first button in the array is selected by default.
You can also mark buttons default with their default flag.
(Read and Write property)

33.7.16   DialogHandle as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the dialog.
Notes: This is only valid (and non zero) while dialog is visible.
(Read only property)

33.7.17   EnableHyperlinks as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to enable hyperlinks.
Notes: Enables hyperlink processing for the strings specified in the Content, ExpandedInformation and Footer members. When enabled, these members may point to strings that contain hyperlinks in the following form:

<A HREF="executablestring">Hyperlink Text</A>

Warning: Enabling hyperlinks when using content from an unsafe source may cause security vulnerabilities.

Note: Task Dialogs will not actually execute any hyperlinks. Hyperlink execution must be handled in the HyperlinkClicked event.
(Read and Write property)

33.7.18   ExpandedByDefault as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates that the string specified by the ExpandedInformation member is displayed when the dialog is initially displayed.
Notes: This flag is ignored if the ExpandedInformation member is empty.

(Read and Write property)

33.7.19 ExpandedControlText as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used to label the button for collapsing the expandable information.
Notes: This member is ignored when the ExpandedInformation member is empty. If this member is empty and the CollapsedControlText is specified, then the CollapsedControlText value will be used for this member as well.
(Read and Write property)

33.7.20 ExpandedInformation as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used for displaying additional information.
Notes: The additional information is displayed either immediately below the content or below the footer text depending on whether the ExpandFooterArea flag is true. If the EnableHyperlinks flag is true, then this string may contain hyperlinks in the form: <A HREF="executablestring">Hyperlink Text</A>.
WARNING: Enabling hyperlinks when using content from an unsafe source may cause security vulnerabilities.
(Read and Write property)

33.7.21 ExpandFooterArea as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether expand footer area is displayed at the bottom.
Notes: Indicates that the string specified by the ExpandedInformation member is displayed at the bottom of the dialog's footer area instead of immediately after the dialog's content. This flag is ignored if the ExpandedInformation member is empty.
(Read and Write property)

33.7.22 Flags as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The flags.
Notes: You should not need to set flags as we have properties for all relevant flags.

(Read and Write property)

33.7.23   Footer as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used in the footer area of the task dialog.
Notes: If EnableHyperlinks is true, this can show clickable links.
(Read and Write property)

33.7.24   FooterIconPicture as Picture

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A picture to be used as Icon that is to be displayed in the footer of the task dialog.
Notes: The picture must have a valid mask.
(Read and Write property)

33.7.25   Icon as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The icon to be displayed in the task dialog.
Notes: Can be one of the kIcon* constants.
Or you set IconPicture property.
(Read and Write property)

33.7.26   IconPicture as Picture

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The icon to be displayed in the task dialog.
Notes: The picture must have a valid mask.
(Read and Write property)

33.7.27   MainInstruction as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used for the main instruction.
Notes: (Read and Write property)

33.7.28   NoDefaultRadioButton as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates that no default item will be selected.
Notes: (Read and Write property)

33.7.29   parent as Variant

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: Parent window.
Notes: Can reference a Window or DesktopWindow object.
(Read and Write property)

33.7.30   parentHandle as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Parent window handle.
Notes: (Read and Write property)

33.7.31   PositionRelativeToWindow as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates that the task dialog is positioned (centered) relative to the window specified by parent.
Notes: If the flag is not supplied (or no parent member is specified), the task dialog is positioned (centered) relative to the monitor.
(Read and Write property)

33.7.32   ProgressbarMax as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum of progress bar.
Notes: Default is 100.

This is a live property. Setting it while dialog is open will change the control.
(Read and Write property)

33.7.33   ProgressbarMin as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The minimum value of the progress bar.
Notes: This is a live property. Setting it while dialog is open will change the control.
(Read and Write property)

33.7.34   ProgressbarState as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The state of the progress bar.
Notes: Default kProgressbarNone.
Can be set to other states to turn on progressbar.
(Read and Write property)

33.7.35   ProgressbarValue as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current value of the progress bar.
Notes: This is a live property. Setting it while dialog is open will change the control.
(Read and Write property)

33.7.36   RightToLeftLayout as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates that text is displayed reading right to left.
Notes: (Read and Write property)

33.7.37   SelectedButton as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID of the clicked button.
Example:

dim td as new TaskDialogMBS

td.CommonButtons = BitwiseOr(td.kCommonButtonOK, td.kCommonButtonCancel)
td.Content = "Hello World"

if td.ShowDialog then

if td.SelectedButton = td.kIDOK then
MsgBox "OK"
else
MsgBox "Cancel"
end if

else
MsgBox "Failed"
end if

Notes: (Read and Write property)

33.7.38   SelectedRadioButton as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID of the selected radio button.
Notes: (Read and Write property)

33.7.39   Showing as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether dialog is currently showing.
Notes: (Read and Write property)

33.7.40   TimedOut as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether we got a timeout.
Notes: (Read only property)

33.7.41   timeoutMS as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The timeout for the dialog.
Notes: In Milliseconds. The dialog closes after given time.
(Read and Write property)

33.7.42   VerificationChecked as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The state of the verification checkbox.
Example:

dim td as new TaskDialogMBS

td.CommonButtons = td.kCommonButtonOK
td.Content = "Hello World"

'td.VerificationChecked = true
td.VerificationText = "Don't show again"

call td.ShowDialog

if td.VerificationChecked then
MsgBox "and we will not show again"
end if

Notes: Before: Indicates whether the verification checkbox in the dialog is checked when the dialog is initially displayed.
After: True if the verification checkbox was checked when the dialog was dismissed.

This is a live property. Changing it while the dialog is open will change the checkbox.
(Read and Write property)

33.7.43   VerificationEnabled as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The enable state of the verification checkbox.
Notes: Can be true to enable the checkbox or false to disable.
(Read and Write property)

33.7.44 VerificationText as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used to label the verification checkbox.
Example:

```
dim td as new TaskDialogMBS

td.CommonButtons = td.kCommonButtonOK
td.Content = "Hello World"

'td.VerificationChecked = true
td.VerificationText = "Don't show again"

call td.ShowDialog

if td.VerificationChecked then
MsgBox "and we will not show again"
end if
```

Notes: If this parameter is empty, the verification checkbox is not displayed in the task dialog. If the VerificationEnabled is false, the checkbox is not enabled.
(Read and Write property)

33.7.45 Width as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the task dialog's client area, in dialog units.
Notes: If 0, the task dialog manager will calculate the ideal width.
(Read and Write property)

33.7.46 WindowTitle as String

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string to be used for the task dialog title.

Notes: (Read and Write property)

33.7.47   Yield as Boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to yield time.
Notes: Whether the plugin should yield time to other Xojo threads while dialog is open.
(Read and Write property)

33.7.48   Button(index as Integer) as TaskDialogButtonMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Array of the custom buttons that are to be displayed in the task dialog.
Notes: You can create TaskDialogButtonMBS objects and assign them to indexes in the array.
(Read and Write computed property)

33.7.49   RadioButton(index as Integer) as TaskDialogButtonMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Array of the custom radio buttons that are to be displayed in the task dialog.
Notes: You can create TaskDialogButtonMBS objects and assign them to indexes in the array.
(Read and Write computed property)

33.7.50   Events

33.7.51   ButtonClicked(ID as Integer) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that a button has been selected.
Notes: The command ID of the button is specified by ID.
To prevent the task dialog from closing, the application must return true, otherwise the task dialog is closed
and the button ID is returned via the original application call.

33.7.52 Close

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the Task Dialog has been destroyed.

33.7.53 Constructed

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the Task Dialog has been created but has not been displayed yet.
Notes: You can adjust here if you need.

33.7.54 ExpandButtonClicked(Expanded as Boolean)

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the expand button has been selected.

33.7.55 Help

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the F1 key has been pressed while the Task Dialog has focus.

33.7.56 HyperlinkClicked(link as string)

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that a hyperlink has been selected.

33.7.57 Navigated

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that navigation has occurred.

33.7.58   Open

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the Task Dialog has been created.

33.7.59   RadioButtonClicked(ID as Integer) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that a radio button has been selected.
Notes: The command ID of the radio button is specified by ID.

33.7.60   Timer(Time as Integer)

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the Task Dialog timer has fired.
Notes: The total elapsed time is specified by Time. You can update the progress bar by setting ProgressbarValue.

This is called regularly so you can do background work.
(or check time and dismiss dialog automatically)

33.7.61   VerificationClicked(Checked as Boolean)

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Indicates that the Task Dialog verification check box has been selected.

33.7.62   Constants

Common Buttons

Constant	Value	Description
kCommonButtonCancel	8	The task dialog contains the push button: Cancel. If this button is specified, the task dialog will respond to typical cancel actions (Alt-F4 and Escape).
kCommonButtonClose	32	The task dialog contains the push button: Close.
kCommonButtonNo	4	The task dialog contains the push button: No.
kCommonButtonOK	1	The task dialog contains the push button: OK.
kCommonButtonRetry	16	The task dialog contains the push button: Retry.
kCommonButtonYes	2	The task dialog contains the push button: Yes.

Icon Constants

Constant	Value	Description
kIconApplication	5	An application icon appears in the task dialog.
kIconError	2	A stop-sign icon appears in the task dialog.
kIconInformation	3	An icon consisting of a lowercase letter i in a circle appears in the task dialog.
kIconNone	0	No icon.
kIconShield	4	A shield icon appears in the task dialog.
kIconWarning	1	An exclamation-point icon appears in the task dialog.

Standard Button IDs

Constant	Value	Description
kIDAbort	3	Abort button
kIDCancel	2	Cancel button
kIDIgnore	5	Ignore button
kIDNo	7	No button
kIDOK	1	OK Button
kIDRetry	4	Retry Button
kIDYes	6	Yes Button

Progressbar State

Constant	Value	Description
kProgressbarError	3	Error state.
kProgressbarMarquee	8	Marquee state
kProgressbarNone	0	No progress bar.
kProgressbarNormal	1	Normal progressbar
kProgressbarPause	2	Paused progress bar

33.8 class TimerMBS

33.8.1 class TimerMBS

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: The class for a timer.
Notes: This timer can work with smaller periods than the normal Xojo timers on Windows.
Works on Mac OS X and Linux, too.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.0

- Unblock menus while threads are running

- MBS Xojo Plugins, version 22.6pr1

- MBS Xojo Plugins, version 20.5pr8

- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1

- MBS Xojo Plugins, version 19.1pr5

- MBS Xojo Plugins, version 17.4pr3

- MBS Xojo / Real Studio Plugins, version 14.4pr1

- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.3

- New Timer for Windows

Xojo Developer Magazine

- 21.3, page 10: News

- 21.2, page 9: News

- 17.3, page 11: News

33.8.2 Methods

33.8.3 Constructor(Period as Integer, Threaded as boolean = true)

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Please pass a period between 1 and 999 Milliseconds.

The timer will try to be as exact as possible, but if main thread is busy, action events are delayed.

Threaded decides on Windows whether you like the high performance version with a thread in background or just a regular timer.

33.8.4   Destructor

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: The destructor.

33.8.5   Properties

33.8.6   Enabled as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: Whether timer is enabled.
Notes: Default is true, but you can set to false to disable the timer for some time.
(Read and Write property)

33.8.7   Period as Integer

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: The period used.
Notes: In milliseconds.
(Read only property)

33.8.8   Threaded as Boolean

Plugin Version: 23.0, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: Whether this timer uses a thread.
Notes: Only true on Windows if you requested a thread.
(Read only property)

33.8.9   Yield as Boolean

Plugin Version: 23.0, Platforms: macOS, Linux, Windows, Targets: Desktop, Console & Web.

Function: Whether we should yield to keep other processes running.

Example:

```
// start timer to yield in order to keep threads running when you use PopupMenus
// in app.open:
#if TargetWindows
Static timer As TimerMBS

timer = New TimerMBS(50, False)
timer.Yield = True
#endif
```

Notes: Setting yield to true tells the plugin to yield time to other threads and timers in the Xojo application.
By yielding with a non threaded TimerMBS, you can keep menus activate on Windows.
(Read and Write property)

33.8.10   Events

33.8.11   Action

Plugin Version: 14.3, Platforms: macOS, Linux, Windows, Targets: .

Function: The action event.

33.9 Globals

33.9.1 GetWindowsErrorMessageMBS(ErrorCode as Integer) as String

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Formats a windows error message for the given error code.
Example:

EditField1.text = GetWindowsErrorMessageMBS(-2147352567)

Notes: Returns empty string for unknown error codes.
Blog Entries

- MBS Real Studio Plugins, version 12.1pr6

33.9.2 InitMessageFilterMBS

Plugin Version: 18.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initializes COM message filter.
Notes: If a COM message can't be handled right away, COM may show a standard dialog to retry, switch task or cancel, which may bring your app in an endless dialog series. MBS Plugin implements a message filter which delays messages till the current processing is done and avoids the dialog. So long running options can perform well.
Blog Entries

- MBS Xojo Plugins 18.3
- Putting an end to the Server Busy Dialog in Xojo and FileMaker
- MBS Xojo Plugins, version 18.3pr2

Videos

- Presentation from Munich conference about MBS Plugins.

33.9.3 WindowsExecuteMBS(ApplicationName as string, CommandLine as string, CurrentDirectory as string, byref PID as integer, Flags as integer = 0, ShowWindow as Integer = -1) as integer

Plugin Version: 11.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new process and its primary thread. The new process runs in the security context of the calling process.
Example:

dim error, pid as Integer

pid = 0

error = WindowsExecuteMBS("", "explorer.exe", "", pid)
if error = 0 then
MsgBox "Launched explorer with process id: "+str(pid)
else
MsgBox "Error: "+str(error)
end if

pid = 0

error = WindowsExecuteMBS("", "notepad.exe ""C:\boot.ini""", "", pid)
if error = 0 then
MsgBox "Launched Notepad with process id: "+str(pid)
else
MsgBox "Error: "+str(error)
end if

Notes: ApplicationName:
The name of the module to be executed. This module can be a Windows-based application. It can be some other type of module (for example, MS-DOS or OS/2) if the appropriate subsystem is available on the local computer.

The string can specify the full path and file name of the module to execute or it can specify a partial name. In the case of a partial name, the function uses the current drive and current directory to complete the specification. The function will not use the search path. This parameter must include the file name extension; no default extension is assumed.

The ApplicationName parameter can be empty. In that case, the module name must be the first white space–delimited token in the CommandLine string. If you are using a long file name that contains a space, use quoted strings to indicate where the file name ends and the arguments begin; otherwise, the file name is ambiguous. For example, consider the string "c:\program files\sub dir\program name". This string can be interpreted in a number of ways. The system tries to interpret the possibilities in the following order:

- c:\program.exe files\sub dir\program name

- c:\program files\sub.exe dir\program name

- c:\program files\sub dir\program.exe name

- c:\program files\sub dir\program name.exe

If the executable module is a 16-bit application, ApplicationName should be empty, and the string pointed

to by CommandLine should specify the executable module as well as its arguments.

To run a batch file, you must start the command interpreter; set ApplicationName to cmd.exe and set CommandLine to the following arguments: /c plus the name of the batch file.

CommandLine:
The command line to be executed. The maximum length of this string is 32,767 characters.. If ApplicationName is empty, the module name portion of CommandLine is limited to MAX_PATH (256) characters.

The CommandLine parameter can be empty. In that case, the function uses the string pointed to by ApplicationName as the command line.

If both ApplicationName and CommandLine are non-empty, the ApplicationName string specifies the module to execute, and the CommandLine string specifies the command line. The new process can use GetCommandLine to retrieve the entire command line. Console processes written in C can use the argc and argv arguments to parse the command line. Because argv [0] is the module name, C programmers generally repeat the module name as the first token in the command line.

If ApplicationName is empty, the first white space–delimited token of the command line specifies the module name. If you are using a long file name that contains a space, use quoted strings to indicate where the file name ends and the arguments begin (see the explanation for the ApplicationName parameter). If the file name does not contain an extension, .exe is appended. Therefore, if the file name extension is .com, this parameter must include the .com extension. If the file name ends in a period (.) with no extension, or if the file name contains a path, .exe is not appended. If the file name does not contain a directory path, the system searches for the executable file in the following sequence:

- The directory from which the application loaded.

- The current directory for the parent process.

- The 32-bit Windows system directory. Use the GetSystemDirectory function to get the path of this directory.

- The 16-bit Windows system directory. There is no function that obtains the path of this directory, but it is searched. The name of this directory is System.

- The Windows directory. Use the GetWindowsDirectory function to get the path of this directory.

- The directories that are listed in the PATH environment variable. Note that this function does not search the per-application path specified by the App Paths registry key. To include this per-application path in the search sequence, use the ShellExecute function.

- The system adds a terminating null character to the command-line string to separate the file name from the arguments. This divides the original string into two strings for internal processing.

CurrentDirectory:
The full path to the current directory for the process. The string can also specify a UNC path.

If this parameter is empty, the new process will have the same current drive and directory as the calling process. (This feature is provided primarily for shells that need to start an application and specify its initial drive and working directory.)

pid:
The variable to store the process ID of the new process.

Returns 0 for success or a windows error code for any error.

Note that the function returns before the process has finished initialization. If a required DLL cannot be located or fails to initialize, the process is terminated. To get the termination status of a process, call GetExitCodeProcess.

Possible Flags:

ShowWindow can be values for ShowWindow function to define how first window of new application is shown. Available values:

Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0

- MBS Xojo Plugins, version 21.0pr8

- MBS REALbasic Plugins, version 11.0pr13

33.9.4 WindowsRunAsMBS(Username as string, Domain as string, Password as string, LoginFlags as Integer, ApplicationName as string, CommandLine as string, CurrentDirectory as string, byref PID as Integer, Flags as Integer = -1) as Integer

Plugin Version: 13.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs an application with a different user login.
Notes: Please see Microsoft website for details:
http://msdn.microsoft.com/en-us/library/windows/desktop/ms682431(v=vs.85).aspx

The plugin passes parameters. If Flags is -1, we use CREATE_DEFAULT_ERROR_MODE Or CREATE_NEW_CONSOLE Or CREATE_NEW_PROCESS_GROUP.
The StartupInfo handles are closed and the PID is returned in the PID parameter.

Returns the Windows error code on failure. Else we return zero for success. On Mac and Linux the result is always -1.

33.9.5 WindowsShellExecuteAsAdminMBS(ParentWindowHandle as integer, File as string, Parameters as string = "", Directory as string = "", ShowCmd as integer = 5) as integer

Plugin Version: 21.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs given file as admin.
Example:

```
// run me as admin
Dim r As Integer = WindowsShellExecuteAsAdminMBS(0, app.ExecutableFile.NativePath, "", "", 3)

If r <>0 Then
MsgBox "error: "+Str(r)
End If
```

Notes: See also WindowsShellExecuteMBS function.
Returns Windows error code, zero on success.

ShowCmd values are shown in documentation WindowsExecuteMBS function.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.1

- MBS Xojo Plugins, version 21.1pr4

Xojo Developer Magazine

- 19.3, page 10: News

33.9.6 WinGetSysColorMBS(Index as Integer) as Color

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the current color of the specified display element.
Notes: Display elements are the parts of a window and the display that appear on the system display screen.

Index: The display element whose color is to be retrieved. This parameter can be one of the following values.

Specifies the left side color in the color gradient of an active window's title bar if the gradient effect is enabled.

Specifies the left side color in the color gradient of an inactive window's title bar if the gradient effect is enabled.

Windows 2000: This value is not supported.

Windows 2000: This value is not supported.

The function returns the red, green, blue (RGB) color value of the given element.

33.9.7   WinOpenFolderAndSelectItemsMBS(folder as folderitem, files() as folderItem, ShowOnDesktop as Boolean = false, EditName as Boolean = false) as Integer

Plugin Version: 14.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens a Windows Explorer window with specified items in a particular folder selected.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim folder as FolderItem = file.Parent

dim r as Integer = WinOpenFolderAndSelectItemsMBS(folder, array(file))

if r = 0 then
MsgBox "OK"
else
MsgBox "Error: "+str(r)
end if

Notes: Please pass a folder to open. Pass the files you want to select in that folder.
Returns Windows error code. Zero for success.

Under Windows XP the flag parameters are ignored. They work in Windows Vista and newer:

Editname: Pass true to select an item and put its name in edit mode. This flag can only be used when a single item is being selected. For multiple item selections, it is ignored.

ShowOnDesktop: Pass true to select the item or items on the desktop rather than in a Windows Explorer

window. Note that if the desktop is obscured behind open windows, it will not be made visible.

For Mac, please use NSWorkspaceMBS.selectfile function.
Blog Entries

- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.4
- MBS Xojo / Real Studio Plugins, version 14.4pr3

33.9.8   WinSetSysColorMBS(Index as Integer, value as Color) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the colors for the specified display elements.
Notes: Display elements are the various parts of a window and the display that appear on the system display screen.
If the function succeeds, the return value is true.
For indexes, please check WinGetSysColorMBS.

33.9.9   WindowsShellExecuteMBS(ParentWindowHandle as Integer, Operation as string, File as string, Parameters as string = "", Directory as string = "", ShowCmd as Integer = 5) as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Performs an operation on a specified file.
Example:

dim e as Integer
dim f as FolderItem

// show documents folder

f = SpecialFolder.Documents
e = WindowsShellExecuteMBS(0, "explore", f.NativePath, "", "", 10)

// Launch Notepad application

e = WindowsShellExecuteMBS(0, "", "notepad.exe", "", "", 10)

// launch file, folder or application

f = SpecialFolder.Desktop.Child("Auto1.JPG")
e = WindowsShellExecuteMBS(0, "", f.NativePath, "", "", 10)

Notes: Use ConsoleExecuteMBS on Mac OS X and Linux.

ParentWindowHandle:
A handle to the owner window used for displaying a UI or error messages. This value can be 0 if the operation is not associated with a window. You can pass window.handle from a Xojo window.

Operation:
A string, referred to in this case as a verb, that specifies the action to be performed. The set of available verbs depends on the particular file or folder. Generally, the actions available from an object's shortcut menu are available verbs. The following verbs are commonly used:

File:
A string that specifies the file or object on which to execute the specified verb. To specify a Shell namespace object, pass the fully qualified parse name. Note that not all verbs are supported on all objects. For example, not all document types support the "print" verb. If a relative path is used for the Directory parameter do not use a relative path for File.

Parameters:
If File specifies an executable file, this parameter is a string that specifies the parameters to be passed to the application. The format of this string is determined by the verb that is to be invoked. If File specifies a document file, Parameters should be "".

Directory:
A string that specifies the default (working) directory for the action. If this value is NULL, the current working directory is used. If a relative path is provided at File, do not use a relative path for Directory.

ShowCmd:
The flags that specify how an application is to be displayed when it is opened. If File specifies a document file, the flag is simply passed to the associated application. It is up to the application to decide how to handle it.

Return Value:
If the function succeeds, it returns a value greater than 32. If the function fails, it returns an error value that indicates the cause of the failure. The return value is cast as an HINSTANCE for backward compatibility with 16-bit Windows applications.

This method allows you to execute any commands in a folder's shortcut menu or stored in the registry. To open a folder, use either of the following calls:

WindowsShellExecuteMBS(handle, "", <fully_qualified_path_to_folder>, "", "", SW_SHOWNORMAL);
or
WindowsShellExecuteMBS(handle, "open", <fully_qualified_path_to_folder>, "", "", SW_SHOWNOR-

MAL);

To explore a folder, use the following call:
WindowsShellExecuteMBS(handle, "explore", <fully_qualified_path_to_folder>, "", "", SW_SHOWNOR-MAL);
To launch the Shell's Find utility for a directory, use the following call.

WindowsShellExecuteMBS(handle, "find", <fully_qualified_path_to_folder>, "", "", 0);

If Operation is empty, the function opens the file specified by File. If Operation is "open" or "explore", the function attempts to open or explore the folder.

Note The Launch folder windows in a separate process setting in Folder Options affects ShellExecute. If that option is disabled (the default setting), ShellExecute uses an open Explorer window rather than launch a new one. If no Explorer window is open, ShellExecute launches a new one.

See also ConsoleExecuteMBS, WindowsShellExecuteAsAdminMBS, WindowsProcessMBS (Windows only), NSTask (Mac only) and ShellMBS (cross platform).

33.9.10   DriveToUNCPathMBS(Driver as string) as string

Plugin Version: 13.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the system tables for which UNC network path belongs to a mapped drive.
Example:

msgbox DriveToUNCPathMBS("W:")

Notes: Please pass the local drive name, which is drive letter and double colon.
Returns empty string on any error.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.2pr5

33.9.11   GetFullWindowsNameMBS(UserName as string, Domain as string) as string

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the full username of an user in a domain.
Example:

dim ShortName as string = "your short name"
dim Domain as string = "your domain"

MsgBox GetFullWindowsNameMBS(ShortName, Domain)

Notes: Returns an empty string on any error.

Use a current Windows NT-style domain, and a username that exists on the PDC.

This will work with any NT-Style PDC (tested on XP bound to Mac OS X Server running SMB as PDC). Windows 2000/2003 and 2008 should also work.

33.10   class WindowsADSystemInfoMBS

33.10.1   class WindowsADSystemInfoMBS

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class retrieves data about the local computer if it is running a Windows operating system in a Windows domain.
Notes: For example, you can get the domain, site, and distinguished name of the local computer.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.3

- MBS Xojo Plugins, version 19.3pr2

Xojo Developer Magazine

- 18.3, page 50: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

- 17.5, page 10: News

33.10.2   Methods

33.10.3   AnyDCName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the DNS name of a domain controller in the local computer's domain.
Notes: Lasterror is set.

33.10.4   Constructor

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Raises unsupportedOperationException on failure.

33.10.5   DCSiteName(Server as String) as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the name of the Active Directory site that contains the local computer.
Notes: Server: DNS name of the service server.

An Active Directory site is one or more well-connected TCP/IP subnets holding Active Directory domain controllers
Lasterror is set.

33.10.6   RefreshSchemaCache

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Refreshes the Active Directory schema cache.
Notes: When you call this method, it does a Put() of the schemaUpdateNow function on the RootDSE. Normally, when you make changes to the schema, they are not updated to the RootDSE until the next automatic update. This method does an immediate update to the schema so that you can view the changes to the schema.
Lasterror is set.

33.10.7   Properties

33.10.8   ComputerName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the distinguished name of the local computer.
Notes: Lasterror is set.
(Read only property)

33.10.9   DomainDNSName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the DNS name of the local computer's domain, such as "domainName.company-Name.com".
Notes: Lasterror is set.
(Read only property)

33.10.10   DomainShortName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the short name of the local computer's domain, such as "domainName".
Notes: Lasterror is set.
(Read only property)

33.10.11   ForestDNSName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the DNS name of the local computer's forest.
Notes: Lasterror is set.
(Read only property)

33.10.12   Handle as Integer

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

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

33.10.13   IsNativeMode as Boolean

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Determines whether the local computer's domain is in native or mixed mode.
Notes: Lasterror is set.
(Read only property)

33.10.14 Lasterror as Integer

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error reported by any of the properties/methods.
Notes: (Read and Write property)

33.10.15 PDCRoleOwner as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the distinguished name of the directory service agent (DSA) object for the DC that owns the primary domain controller role in the local computer's domain.
Notes: Lasterror is set.
(Read only property)

33.10.16 SchemaRoleOwner as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the distinguished name of the directory service agent (DSA) object for the DC that owns the schema master role in the local computer's forest.
Notes: Lasterror is set.
(Read only property)

33.10.17 SiteName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the site name of the local computer.
Notes: Lasterror is set.
(Read only property)

33.10.18 UserName as String

Plugin Version: 19.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the Active Directory distinguished name of the current user, which is the logged-on user or the user impersonated by the calling thread.
Notes: Lasterror is set.
(Read only property)

Constant	Value	Description
CREATE_BREAKAWAY_FROM_JOB	&h01000000	The child processes of a process associated with a job are not associated with the job. If the calling process is not associated with a job, this constant has no effect. If the calling process is associated with a job, the job must set the JOB_OBJECT_LIMIT_BREAKAWAY_OK limit.
CREATE_DEFAULT_ERROR_MODE	&h04000000	The new process does not inherit the error mode of the calling process. Instead, the new process gets the default error mode. This feature is particularly useful for multi-threaded shell applications that run with hard errors disabled. The default behavior is for the new process to inherit the error mode of the caller. Setting this flag changes that default behavior.
CREATE_NEW_CONSOLE	&h00000010	The new process has a new console, instead of inheriting its parent's console (the default). For more information, see Creation of a Console. This flag cannot be used with DETACHED_PROCESS.
CREATE_NEW_PROCESS_GROUP	&h00000200	The new process is the root process of a new process group. The process group includes all processes that are descendants of this root process. The process identifier of the new process group is the same as the process identifier, which is returned in the lpProcessInformation parameter. Process groups are used by the GenerateConsoleCtrlEvent function to enable sending a CTRL+BREAK signal to a group of console processes. If this flag is specified, CTRL+C signals will be disabled for all processes within the new process group. This flag is ignored if specified with CREATE_NEW_CONSOLE.
CREATE_NO_WINDOW	&h08000000	The process is a console application that is being run without a console window. Therefore, the console handle for the application is not set. This flag is ignored if the application is not a console application, or if it is used with either CREATE_NEW_CONSOLE or DETACHED_PROCESS.
CREATE_PROTECTED_PROCESS	&h00040000	The process is to be run as a protected process. The system restricts access to protected processes and the threads of protected processes. For more information on how processes can interact with protected processes, see Process Security and Access Rights. To activate a protected process, the binary must have a special signature. This signature is provided by Microsoft but not currently available for non-Microsoft binaries. There are currently four protected processes: media foundation, audio engine, Windows error reporting, and system. Components that load into these binaries must also be signed. Multimedia companies can leverage the first two protected processes. For more information, see Overview of the Protected Media Path. Windows Server 2003 and Windows XP/2000: This value is not supported.
CREATE_PRESERVE_CODE_AUTHZ_LEVEL	&h02000000	Allows the caller to execute a child process that bypasses the process restrictions that would normally be applied automatically to the process. Windows 2000: This value is not supported.
CREATE_SEPARATE_WOW_VDM	&h00000800	This flag is valid only when starting a 16-bit Windows-based application. If set, the new process runs in a private Virtual DOS Machine (VDM). By default, all 16-bit Windows-based applications run as threads in a single, shared VDM. The advantage of running separately is that a crash only terminates the single VDM; any other programs running in distinct VDMs continue to function normally. Also, 16-bit Windows-based applications that are run in separate VDMs have separate input queues. That means that if one application stops responding momentarily, applications in separate VDMs continue to receive input. The disadvantage of running separately is that it takes significantly more memory to do so. You should use this flag only if the user requests that 16-bit applications should run in their own VDM.
CREATE_SHARED_WOW_VDM	&h00001000	The flag is valid only when starting a 16-bit Windows-based application. If the DefaultSeparateVDM switch in the Windows section of WIN.INI is TRUE, this flag overrides the switch. The new process is run in the shared Virtual DOS Machine.
CREATE_SUSPENDED	&h00000004	The primary thread of the new process is created in a suspended state, and does not run until the ResumeThread function is called.
CREATE_UNICODE_ENVIRONMENT	&h00000400	If this flag is set, the environment block pointed to by lpEnvironment uses Unicode characters. Otherwise, the environment block uses ANSI characters.
DEBUG_ONLY_THIS_PROCESS	&h00000002	The calling thread starts and debugs the new process. It can receive all related debug events using the WaitForDebugEvent function.
DEBUG_PROCESS	&h00000001	The calling thread starts and debugs the new process and all child processes created by the new process. It receive all related debug events using the WaitForDebugEvent function. A process that uses DEBUG_PROCESS becomes the root of a debugging chain. This continues until another process in the chain is created with DEBUG_PROCESS. If this flag is combined with DEBUG_ONLY_THIS_PROCESS, the caller debugs only the new process, not any child processes.
DETACHED_PROCESS	&h00000008	For console processes, the new process does not inherit its parent's console (the default). The new process can call the AllocConsole function at a later time to create a console. For more information, see Creation of a Console. This value cannot be used with CREATE_NEW_CONSOLE.
EXTENDED_STARTUPINFO_PRESENT	&h00080000	The process is created with extended startup information; the lpStartupInfo parameter specifies a STARTUPINFOEX structure. Windows Server 2003 and Windows XP/2000: This value is not supported.
INHERIT_PARENT_AFFINITY	&h00010000	The process inherits its parent's affinity. If the parent process has threads in more than one processor group, the new process inherits the group-relative affinity of an arbitrary group in use by the parent. Windows Server 2008, Windows Vista, Windows Server 2003, and Windows XP/2000: This value is not supported.
ABOVE_NORMAL_PRIORITY_CLASS	&h00008000	Process that has priority above NORMAL_PRIORITY_CLASS but below HIGH_PRIORITY_CLASS.
BELOW_NORMAL_PRIORITY_CLASS	&h00004000	Process that has priority above IDLE_PRIORITY_CLASS but below NORMAL_PRIORITY_CLASS.

Constants	Value	Meaning
SW_FORCEMINIMIZE	11	Minimizes a window, even if the thread that owns the window is not responding. This flag should only be used when minimizing windows from a different thread.
SW_HIDE	0	Hides the window and activates another window.
SW_MAXIMIZE	3	Maximizes the specified window.
SW_MINIMIZE	6	Minimizes the specified window and activates the next top-level window in the Z order.
SW_RESTORE	9	Activates and displays the window. If the window is minimized or maximized, the system restores it to its original size and position. An application should specify this flag when restoring a minimized window.
SW_SHOW	5	Activates the window and displays it in its current size and position.
SW_SHOWDEFAULT	10	Sets the show state based on the SW_ value specified in the STARTUPINFO structure passed to the CreateProcess function by the program that started the application.
SW_SHOWMAXIMIZED	3	Activates the window and displays it as a maximized window.
SW_SHOWMINIMIZED	2	Activates the window and displays it as a minimized window.
SW_SHOWMINNOACTIVE	7	Displays the window as a minimized window. This value is similar to SW_SHOWMINIMIZED, except the window is not activated.
SW_SHOWNA	8	Displays the window in its current size and position. This value is similar to SW_SHOW, except that the window is not activated.
SW_SHOWNOACTIVATE	4	Displays a window in its most recent size and position. This value is similar to SW_SHOWNORMAL, except that the window is not activated.
SW_SHOWNORMAL	1	Activates and displays a window. If the window is minimized or maximized, the system restores it to its original size and position. An application should specify this flag when displaying the window for the first time.

Name	Value	Meaning
COLOR_3DDKSHADOW	21	Dark shadow for three-dimensional display elements.
COLOR_3DFACE	15	Face color for three-dimensional display elements and for dialog box backgrounds.
COLOR_3DHIGHLIGHT	20	Highlight color for three-dimensional display elements (for edges facing the light source.)
COLOR_3DHILIGHT	20	Highlight color for three-dimensional display elements (for edges facing the light source.)
COLOR_3DLIGHT	22	Light color for three-dimensional display elements (for edges facing the light source.)
COLOR_3DSHADOW	16	Shadow color for three-dimensional display elements (for edges facing away from the light source).
COLOR_ACTIVEBORDER	10	Active window border.
COLOR_ACTIVECAPTION	2	Active window title bar.

COLOR_APPWORKSPACE	12	Background color of multiple document interface (MDI) applications.
COLOR_BACKGROUND	1	Desktop.
COLOR_BTNFACE	15	Face color for three-dimensional display elements and for dialog box backgrounds.
COLOR_BTNHIGHLIGHT	20	Highlight color for three-dimensional display elements (for edges facing the light source.)
COLOR_BTNHILIGHT	20	Highlight color for three-dimensional display elements (for edges facing the light source.)
COLOR_BTNSHADOW	16	Shadow color for three-dimensional display elements (for edges facing away from the light source).
COLOR_BTNTEXT	18	Text on push buttons.
COLOR_CAPTIONTEXT	9	Text in caption, size box, and scroll bar arrow box.
COLOR_DESKTOP	1	Desktop.
COLOR_GRADIENTACTIVECAPTION	27	Right side color in the color gradient of an active window's title bar. COLOR_ACTIVECAPTION specifies the left side color. Use SPI_GET-GRADIENTCAPTIONS with the SystemParametersInfo function to determine whether the gradient effect is enabled.
COLOR_GRADIENTINACTIVECAPTION	28	Right side color in the color gradient of an inactive window's title bar. COLOR_INACTIVECAPTION specifies the left side color.
COLOR_GRAYTEXT	17	Grayed (disabled) text. This color is set to 0 if the current display driver does not support a solid gray color.
COLOR_HIGHLIGHT	13	Item(s) selected in a control.
COLOR_HIGHLIGHTTEXT	14	Text of item(s) selected in a control.
COLOR_HOTLIGHT	26	Color for a hyperlink or hot-tracked item.
COLOR_INACTIVEBORDER	11	Inactive window border.
COLOR_INACTIVECAPTION	3	Inactive window caption.

COLOR_INACTIVECAPTIONTEXT	19	Color of text in an inactive caption.
COLOR_INFOBK	24	Background color for tooltip controls.
COLOR_INFOTEXT	23	Text color for tooltip controls.
COLOR_MENU	4	Menu background.
COLOR_MENUHILIGHT	29	The color used to highlight menu items when the menu appears as a flat menu (see SystemParametersInfo). The highlighted menu item is outlined with COLOR_HIGHLIGHT.

COLOR_MENUBAR	30	The background color for the menu bar when menus appear as flat menus (see SystemParametersInfo). However, COLOR_MENU continues to specify the background color of the menu popup.

COLOR_MENUTEXT	7	Text in menus.
COLOR_SCROLLBAR	0	Scroll bar gray area.
COLOR_WINDOW	5	Window background.
COLOR_WINDOWFRAME	6	Window frame.
COLOR_WINDOWTEXT	8	Text in windows.

edit	Launches an editor and opens the document for editing. If File is not a document file, the function will fail.
explore	Explores a folder specified by File.
find	Initiates a search beginning in the directory specified by Directory.
open	Opens the item specified by the File parameter. The item can be a file or folder.
print	Prints the file specified by File. If File is not a document file, the function fails.
””	In systems prior to Windows 2000, the default verb is used if it is valid and available in the registry. If not, the ”open” verb is used. In Windows 2000 and later, the default verb is used if available. If not, the ”open” verb is used. If neither verb is available, the system uses the first verb listed in the registry.

SW_HIDE	0	Hides the window and activates another window.
SW_MAXIMIZE	3	Maximizes the specified window.
SW_MINIMIZE	6	Minimizes the specified window and activates the next top-level window in the z-order.
SW_RESTORE	9	Activates and displays the window. If the window is minimized or maximized, Windows restores it to its original size and position. An application should specify this flag when restoring a minimized window.
SW_SHOW	5	Activates the window and displays it in its current size and position.
SW_SHOWDEFAULT	10	Sets the show state based on the SW_ flag specified in the STARTUPINFO structure passed to the CreateProcess function by the program that started the application. An application should call ShowWindow with this flag to set the initial show state of its main window.
SW_SHOWMAXIMIZED	3	Activates the window and displays it as a maximized window.
SW_SHOWMINIMIZED	2	Activates the window and displays it as a minimized window.
SW_SHOWMINNOACTIVE	7	Displays the window as a minimized window. The active window remains active.
SW_SHOWNA	8	Displays the window in its current state. The active window remains active.
SW_SHOWNOACTIVATE	4	Displays a window in its most recent size and position. The active window remains active.
SW_SHOWNORMAL	1	Activates and displays a window. If the window is minimized or maximized, Windows restores it to its original size and position. An application should specify this flag when displaying the window for the first time.

Return code	Description
0	The operating system is out of memory or resources.
ERROR_FILE_NOT_FOUND = 2	The specified file was not found.
ERROR_PATH_NOT_FOUND = 3	The specified path was not found.
ERROR_BAD_FORMAT = 11	The .exe file is invalid (non-Win32 .exe or error in .exe image).
SE_ERR_ACCESSDENIED = 5	The operating system denied access to the specified file.
SE_ERR_ASSOCINCOMPLETE = 27	The file name association is incomplete or invalid.
SE_ERR_DDEBUSY = 30	The DDE transaction could not be completed because other DDE transactions were being processed.
SE_ERR_DDEFAIL = 29	The DDE transaction failed.
SE_ERR_DDETIMEOUT = 28	The DDE transaction could not be completed because the request timed out.
SE_ERR_DLLNOTFOUND = 32	The specified DLL was not found.
SE_ERR_FNF = 2	The specified file was not found.
SE_ERR_NOASSOC = 31	There is no application associated with the given file name extension. This error will also be returned if you attempt to print a file that is not printable.
SE_ERR_OOM = 8	There was not enough memory to complete the operation.
SE_ERR_PNF = 3	The specified path was not found.
SE_ERR_SHARE = 26	A sharing violation occurred.

33.11 class WindowsClipboardMBS

33.11.1 class WindowsClipboardMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The windows clipboard class.
Notes: Use it to access the clipboard on Windows.

Be aware that in the constructor we open the clipboard connection. Since Xojo may itself like to look into the clipboard to decide whether to enable copy & paste events, our clipboard connection may get taken or times out. Only one app at a time can have clipboard access.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.4

- MBS Xojo Plugins, version 17.4pr5

- MBS Xojo Plugins, version 17.4pr3

- Windows, Xojo and the Clipboard

- MBS Xojo / Real Studio Plugins, version 16.5pr4

- MBS Xojo / Real Studio Plugins, version 15.1pr7

- [ANN] MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 14.4

- New MBS REALbasic Plugin Version 10.4

- MBS REALbasic Plugins Version 10.4 release notes

- Teaser: Clipboard classes

Videos

- Presentation from Munich conference about MBS Plugins.

Xojo Developer Magazine

- 15.6, page 9: News

33.11.2 Methods

33.11.3 Clear

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Clears the contents of the clipboard.

33.11.4   ClipboardFormats as Integer()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an array with all clipboard formats currently in the clipboard.

33.11.5   ClipboardSequenceNumber as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the clipboard sequence number for the current window station.
Notes: The return value is the clipboard sequence number. If you do not have WINSTA_ACCESSCLIP-BOARD access to the window station, the function returns zero.

The system keeps a serial number for the clipboard for each window station. This number is incremented whenever the contents of the clipboard change or the clipboard is emptied. You can track this value to determine whether the clipboard contents have changed and optimize creating DataObjects. If clipboard rendering is delayed, the sequence number is not incremented until the changes are rendered.

33.11.6   Constructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Connects to the clipboard.
Notes: As the clipboard has exclusive access, do not keep WindowsClipboardMBS objects around. Best you only use them within a method so once the method ends, everything is released.

33.11.7   CountClipboardFormats as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the number of different data formats currently on the clipboard.
Notes: If the function succeeds, the return value is the number of different data formats currently on the clipboard.

If the function fails, the return value is zero.

33.11.8   Destructor

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Automatically closes the clipboard.

33.11.9 EnumClipboardFormats(format as Integer = 0) as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enumerates the data formats currently available on the clipboard.
Notes: Clipboard data formats are stored in an ordered list. To perform an enumeration of clipboard data formats, you make a series of calls to the EnumClipboardFormats function. For each call, the format parameter specifies an available clipboard format, and the function returns the next available clipboard format.

format: A clipboard format that is known to be available.

To start an enumeration of clipboard formats, set format to zero. When format is zero, the function retrieves the first available clipboard format. For subsequent calls during an enumeration, set format to the result of the previous EnumClipboardFormats call.

If the function succeeds, the return value is the clipboard format that follows the specified format, namely the next available clipboard format.

If the function fails, the return value is zero.
If there are no more clipboard formats to enumerate, the return value is zero.

You must open the clipboard before enumerating its formats. Use the OpenClipboard function to open the clipboard. The EnumClipboardFormats function fails if the clipboard is not open.

The EnumClipboardFormats function enumerates formats in the order that they were placed on the clipboard. If you are copying information to the clipboard, add clipboard objects in order from the most descriptive clipboard format to the least descriptive clipboard format. If you are pasting information from the clipboard, retrieve the first clipboard format that you can handle. That will be the most descriptive clipboard format that you can handle.

The system provides automatic type conversions for certain clipboard formats. In the case of such a format, this function enumerates the specified format, then enumerates the formats to which it can be converted. For more information, see Standard Clipboard Formats and Synthesized Clipboard Formats.

33.11.10 GetClipboardFormatName(format as Integer) as string

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves from the clipboard the name of the specified registered format. The function copies the name to the specified buffer.
Example:

```
// show items on clipboard

dim w as new WindowsClipboardMBS
dim types(-1) as integer = w.ClipboardFormats
dim names() as string

for each type as integer in types
dim n as string = w.GetClipboardFormatName(type)

n = str(type)+" "+n

names.append n
next

MsgBox Join(names,EndOfLine)
```

33.11.11   GetData(type as Integer) as string

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves data from the clipboard in a specified format.
Notes: Format: A clipboard format.

If the function succeeds, the return value is the string with the clipboard data in the specified format.

If the function fails, the return value is "".

An application can enumerate the available formats in advance by using the EnumClipboardFormats function.

33.11.12   GetDIB as Picture

Plugin Version: 17.4, Platform: Windows, Targets: Desktop only.

Function: Queries DIB picture on the clipboard.
Notes: Returns nil on any error.
This is using DIB type in clipboard for a Device Independent Bitmap.

Please use GetPicture for bitmap handles.

33.11.13   GetFiles as string()

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries list of file paths on the clipboard.
Notes: This is for working with Explorers copy & paste feature.

33.11.14   GetPicture as Picture

Plugin Version: 14.4, Platform: Windows, Targets: Desktop only.

Function: Queries bitmap on the clipboard.
Notes: Returns nil on any error.
This is using BITMAP type in clipboard for a bitmap handle.

Please use GetDIB for Device Independent Bitmap.

33.11.15   IsClipboardFormatAvailable(type as Integer) as boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Determines whether the clipboard contains data in the specified format.
Notes: type: A standard or registered clipboard format.

If the clipboard format is available, the return value is true.
If the clipboard format is not available, the return value is false.

Typically, an application that recognizes only one clipboard format would call this function when processing the EnableMenuItems event. The application would then enable or disable the Paste menu item, depending on the return value.

33.11.16   RegisterClipboardFormat(type as string) as Integer

Plugin Version: 13.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers a new clipboard format. This format can then be used as a valid clipboard format.
Example:

dim TypeHTML as Integer
dim TypeRTF as Integer

TypeHTML = WindowsClipboardMBS.RegisterClipboardFormat("HTML Format")
TypeRTF = WindowsClipboardMBS.RegisterClipboardFormat("Rich Text Format")

Notes: If the function succeeds, the return value identifies the registered clipboard format.
If the function fails, the return value is zero.

If a registered format with the specified name already exists, a new format is not registered and the return value identifies the existing format. This enables more than one application to copy and paste data using the same registered clipboard format. Note that the format name comparison is case-insensitive.
Registered clipboard formats are identified by values in the range &hC000 through &hFFFF.

33.11.17 SetData(type as Integer, rawData as string) as boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Places data on the clipboard in a specified clipboard format.
Notes: Format: The clipboard format. This parameter can be a registered format or any of the standard clipboard formats.
rawData: A string with the data in the specified format.

If the function succeeds, the return value is true.
If the function fails, the return value is false.

The system performs implicit data format conversions between certain clipboard formats when an application calls the GetClipboardData function. For example, if the kTypeOEMTEXT format is on the clipboard, a window can retrieve data in the kTypeTEXT format. The format on the clipboard is converted to the requested format on demand. For more information, see Synthesized Clipboard Formats.

33.11.18 SetDIB(pic as Picture) as boolean

Plugin Version: 17.4, Platform: Windows, Targets: Desktop only.

Function: Puts a DIB picture on the clipboard.
Notes: Returns true on success.
This is using DIB type in clipboard for a bitmap handle.

Please use SetPicture for Device Independent Bitmap.

33.11.19 SetFiles(paths() as string) as boolean

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Puts a list of file paths on the clipboard.
Notes: This is for working with Explorers copy & paste feature.
Returns true on success.
Folder paths may work better if they have no training backslash.

33.11.20 SetPicture(pic as Picture) as boolean

Plugin Version: 14.4, Platform: Windows, Targets: Desktop only.

Function: Puts a bitmap picture on the clipboard.
Notes: Returns true on success.
This is using BITMAP type in clipboard for a bitmap handle.

Please use SetDIB for Device Independent Bitmap.

33.11.21 Properties

33.11.22 Valid as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A flag whether the constructor got a clipboard connection.
Notes: (Read and Write property)

33.11.23 Constants

Constants

Constant	Value	Description
kTypeBitmap	2	One of standard clipboard formats defined by the system. A handle to a bitmap (HBITMAP).
kTypeDIB	8	One of standard clipboard formats defined by the system. A memory object containing a BITMAPINFO structure followed by the bitmap bits.
kTypeDIBV5	17	One of standard clipboard formats defined by the system. A memory object containing a BITMAPV5HEADER structure followed by the bitmap color space information and the bitmap bits.
kTypeDIF	5	One of standard clipboard formats defined by the system. Software Arts' Data Interchange Format.
kTypeENHMetaFile	14	One of standard clipboard formats defined by the system. A handle to an enhanced metafile (HENHMETAFILE).
kTypeHDROP	15	One of standard clipboard formats defined by the system. A handle to type HDROP that identifies a list of files. An application can retrieve information about the files by passing the handle to the DragQueryFile function.
kTypeLocale	16	One of standard clipboard formats defined by the system. The data is a handle to the locale identifier associated with text in the clipboard. When you close the clipboard, if it contains kTypeText data but no kTypeLocale data, the system automatically sets the kTypeLocale format to the current input language. You can use the kTypeLocale format to associate a different locale with the clipboard text. An application that pastes text from the clipboard can retrieve this format to determine which character set was used to generate the text. Note that the clipboard does not support plain text in multiple character sets. To achieve this, use a formatted text data type such as RTF instead. The system uses the code page associated with kTypeLocale to implicitly convert from kTypeText to kTypeUnicodeText. Therefore, the correct code page table is used for the conversion.
kTypeMetaFilePict	3	One of standard clipboard formats defined by the system. Handle to a metafile picture format as defined by the METAFILEPICT structure. When passing a kTypeMetaFilePict handle by means of DDE, the application responsible for deleting hMem should also free the metafile referred to by the kTypeMetaFilePict handle.
kTypeOEMText	7	One of standard clipboard formats defined by the system. Text format containing characters in the OEM character set. Each line ends with a carriage return/linefeed (CR-LF) combination. A null character signals the end of the data.
kTypePalette	9	One of standard clipboard formats defined by the system. Handle to a color palette. Whenever an application places data in the clipboard that depends on or assumes a color palette, it should place the palette on the clipboard as well. If the clipboard contains data in the kTypePalette (logical color palette) format, the application should use the SelectPalette and RealizePalette functions to realize (compare) any other data in the clipboard against that logical palette. When displaying clipboard data, the clipboard always uses as its current palette any object on the clipboard that is in the kTypePalette format.
kTypePenData	10	One of standard clipboard formats defined by the system. Data for the pen extensions to the Microsoft Windows for Pen Computing
kTypeRIFF	11	One of standard clipboard formats defined by the system. Represents audio data more complex than can be represented in a CF_WAVE standard wave format.
kTypeSylk	4	One of standard clipboard formats defined by the system. Microsoft Symbolic Link (SYLK) format.
kTypeText	1	One of standard clipboard formats defined by the system. Text format. Each line ends with a carriage return/linefeed (CR-LF) combination. A null character signals the end of the data. Use this format for ANSI text.

33.12 class WindowsDeviceMBS

33.12.1 class WindowsDeviceMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class is for device
Example:

dim devices(-1) as WindowsDeviceMBS = WindowsDeviceMBS.Devices
for each device as WindowsDeviceMBS in devices
msgbox device.Description
next

Notes: For Mac you can take a look on the MacUSBDeviceMBS class and the IORegistryMBS class.
See also WinUSBDeviceMBS and WinUSBMBS classes.
For cross platform development also check LibUSBDeviceMBS classes.
Blog Entries

- MBS Real Studio Plugins, version 11.2pr7

- New MBS REALbasic Plugin Version 10.4

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr11

- MacUSBDeviceMBS

- MBS REALbasic Plugins, version 10.4pr4

33.12.2 Methods

33.12.3 CompatibleIDs as string()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The list of compatible IDs for a device.
Notes: A device has only one device ID that is the most specific ID for a device. A device can also have less specific hardware IDs and compatible IDs. Windows first tries to find an INF file that matches the device ID. If it does not find such a match, Windows next tries to find a match to one of the less specific hardware IDs or compatible IDs.

33.12.4   Devices(ClassGUID as string, present as boolean = true) as Windows-DeviceMBS()

Plugin Version: 11.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all devices on the system matching the given Class GUID.
Example:

const PortsGUID = " { 4D36E978-E325-11CE-BFC1-08002BE10318 } "
dim devices(-1) as WindowsDeviceMBS = WindowsDeviceMBS.Devices(PortsGUID)

Notes: If present is true, only devices currently present on the system are returned.
The array is empty on any error.

Class	{ GUID }	Device Description
CDROM	{ 4D36E965-E325-11CE-BFC1-08002BE10318 }	CD/DVD/Blu-ray drives
DiskDrive	{ 4D36E967-E325-11CE-BFC1-08002BE10318 }	Hard drives
Display	{ 4D36E968-E325-11CE-BFC1-08002BE10318 }	Video adapters
FDC	{ 4D36E969-E325-11CE-BFC1-08002BE10318 }	Floppy controllers
FloppyDisk	{ 4D36E980-E325-11CE-BFC1-08002BE10318 }	Floppy drives
HDC	{ 4D36E96A-E325-11CE-BFC1-08002BE10318 }	Hard drive controllers
HIDClass	{ 745A17A0-74D3-11D0-B6FE-00A0C90F57DA }	Some USB devices
1394	{ 6BDD1FC1-810F-11D0-BEC7-08002BE2092F }	IEEE 1394 host controller
Image	{ 6BDD1FC6-810F-11D0-BEC7-08002BE2092F }	Cameras and scanners
Keyboard	{ 4D36E96B-E325-11CE-BFC1-08002BE10318 }	Keyboards
Modem	{ 4D36E96D-E325-11CE-BFC1-08002BE10318 }	Modems
Mouse	{ 4D36E96F-E325-11CE-BFC1-08002BE10318 }	Mice and pointing devices
Media	{ 4D36E96C-E325-11CE-BFC1-08002BE10318 }	Audio and video devices
Net	{ 4D36E972-E325-11CE-BFC1-08002BE10318 }	Network adapters
Ports	{ 4D36E978-E325-11CE-BFC1-08002BE10318 }	Serial and parallel ports
SCSIAdapter	{ 4D36E97B-E325-11CE-BFC1-08002BE10318 }	SCSI and RAID controllers
System	{ 4D36E97D-E325-11CE-BFC1-08002BE10318 }	System buses, bridges, etc.
USB	{ 36FC9E60-C465-11CF-8056-444553540000 }	USB host controllers and hubs

See also:

- 33.12.5 Devices(present as boolean = true) as WindowsDeviceMBS()   1048

33.12.5   Devices(present as boolean = true) as WindowsDeviceMBS()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all devices on the system.
Notes: If present is true, only devices currently present on the system are returned.
The array is empty on any error.
See also:

- 33.12.4 Devices(ClassGUID as string, present as boolean = true) as WindowsDeviceMBS()   1048

33.12.6 HardwareID as string()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The list of hardware IDs for a device.

33.12.7 LocationPaths as string()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The location of the device in the device tree.
Notes: Windows Server 2003 and later.

33.12.8 LowerFilters as string()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The names of a device's lower-filter drivers.

33.12.9 UpperFilters as string()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The names of a device's upper filter drivers.

33.12.10 Properties

33.12.11 Address as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's address.
Notes: (Read and Write property)

33.12.12 BusNumber as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's bus number.
Notes: (Read and Write property)

33.12.13   BusTypeGUID as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GUID for the device's bus type.
Notes: (Read and Write property)

33.12.14   Capabilities as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device capabilities.
Notes: A combination of the kDeviceCapability* constants.
(Read and Write property)

33.12.15   Characteristics as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device characteristics.
Notes: (Read and Write property)

33.12.16   ClassGUID as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GUID that represents the device setup class of a device.
Notes: (Read and Write property)

33.12.17   ClassName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device setup class of a device.
Notes: (This property should be named Class, but was renamed ClassName as Class is a reserved word in Xojo)

(Read and Write property)

33.12.18   ConfigFlags as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's configuration flags.
Notes: (Read and Write property)

33.12.19   Description as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The description of a device.
Notes: Maybe localized.
(Read and Write property)

33.12.20   DeviceID as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: If a device is associated with this device entry, here you have the ID string for this device.
Notes: (Read and Write property)

33.12.21   DevicePath as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: For devices with device ID, this is the device path.
Notes: (Read and Write property)

33.12.22   DeviceType as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's type.
Notes: See kDeviceType* constants.
(Read and Write property)

33.12.23   Driver as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that identifies the device's software key (sometimes called the driver key).
Notes: (Read and Write property)

33.12.24   EnumeratorName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The string that contains the name of the device's enumerator.
Notes: (Read and Write property)

33.12.25   Exclusive as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether a user can obtain exclusive use of the device.
Notes: The value is true if exclusive use is allowed, or false otherwise.
(Read and Write property)

33.12.26   FriendlyName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The friendly name of a device.
Notes: (Read and Write property)

33.12.27   HID as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the plugin found additional HID details on the device.
Notes: (Read and Write property)

33.12.28   HIDAccessible as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the plugin was able to access the device and query the properties.
Notes: HIDAccessible is true if and only if HID is also true.
(Read and Write property)

33.12.29   HIDFeatureReportByteLength as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The length of the HID feature reports for this USB HID device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.30   HIDInputReportByteLength as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The length of the HID input reports for this USB HID device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.31   HIDManufacturerName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The manufacturer name for this device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.32   HIDOutputReportByteLength as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The length of the HID output reports for this USB HID device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.33   HIDProductID as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The product ID for this USB device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.34   HIDProductName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The product name for this HID Device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.35   HIDSerialNumber as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The serial number for this USB device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.36   HIDVendorID as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The vendor ID for this USB device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.37   HIDVersionNumber as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The version number for this HID Device.
Notes: Only valid if HID and HIDAccessible are true.
(Read and Write property)

33.12.38   InstallState as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The install state of this device.
Notes: See kInstallState* constants.
(Read and Write property)

33.12.39   LegacyBusType as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's legacy bus type.
Notes: (Read and Write property)

33.12.40   LocationInformation as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The hardware location of a device.
Notes: (Read and Write property)

33.12.41   Manufacturer as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the device manufacturer.
Notes: (Read and Write property)

33.12.42   PhysicalDeviceObjectName as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name that is associated with the device's PDO.
Notes: (Read and Write property)

33.12.43   RemovalPolicy as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's current removal policy.
Notes: Windows XP and later
(Read and Write property)

33.12.44 RemovalPolicyHWDefault as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's hardware-specified default removal policy.
Notes: Windows XP and later
(Read and Write property)

33.12.45 RemovalPolicyOverride as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's override removal policy (if it exists) from the registry.
Notes: Windows XP and later
(Read and Write property)

33.12.46 SecurityDescriptor as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device's security descriptor.
Notes: The format of security descriptor strings is described in Microsoft Windows SDK documentation.
(Read and Write property)

33.12.47 Service as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The service name for a device.
Notes: (Read and Write property)

33.12.48 UINumber as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The UINumber.
Notes: Specifies a number associated with the device that can be displayed in the user interface.

This number is typically a user-perceived slot number, such as a number printed next to the slot on the board, or some other number that makes locating the physical device easier for the user. For buses with no such convention, or when the UINumber is unknown, the bus driver leaves this member at its default value

of -1.
(Read and Write property)

33.12.49 Constants

Constants

Constant	Value	Description
kDeviceCapabilityDockDevice	8	One of the capabilities constants. Dock Device
kDeviceCapabilityEjectSupported	2	One of the capabilities constants. Eject Supported
kDeviceCapabilityHardwareDisabled	256	One of the capabilities constants. Hardware Disabled
kDeviceCapabilityLockSupported	1	One of the capabilities constants. Lock Supported
kDeviceCapabilityNonDynamic	512	One of the capabilities constants. Non Dynamic
kDeviceCapabilityRAWDeviceOK	64	One of the capabilities constants. Raw Device OK
kDeviceCapabilityRemovable	4	One of the capabilities constants. Removable
kDeviceCapabilitySilentInstall	32	One of the capabilities constants. Silent Install
kDeviceCapabilitySurpriseRemovalOK	128	One of the capabilities constants. Surprise Removal OK
kDeviceCapabilityUniqueID	16	One of the capabilities constants. Unique ID
kDeviceType8042Port	&h27	One of the device type constants.
kDeviceTypeACPI	&h32	One of the device type constants.
kDeviceTypeBattery	&h29	One of the device type constants.
kDeviceTypeBeep	&h01	One of the device type constants.
kDeviceTypeBusExtender	&h2a	One of the device type constants.
kDeviceTypeCDROM	&h02	One of the device type constants.
kDeviceTypeCDROWFileSystem	&h03	One of the device type constants.
kDeviceTypeChanger	&h30	One of the device type constants.
kDeviceTypeController	&h04	One of the device type constants.
kDeviceTypeDataLink	&h05	One of the device type constants.
kDeviceTypeDFS	&h06	One of the device type constants.
kDeviceTypeDFSFileSystem	&h35	One of the device type constants.
kDeviceTypeDFSVolume	&h36	One of the device type constants.
kDeviceTypeDisk	&h07	One of the device type constants.
kDeviceTypeDiskFileSystem	&h08	One of the device type constants.
kDeviceTypeDVD	&h33	One of the device type constants.
kDeviceTypeFileSystem	&h09	One of the device type constants.
kDeviceTypeFips	&h3a	One of the device type constants.
kDeviceTypeFullscreenVideo	&h34	One of the device type constants.
kDeviceTypeInportPort	&h0a	One of the device type constants.
kDeviceTypeKeyboard	&h0b	One of the device type constants.
kDeviceTypeKS	&h2f	One of the device type constants.
kDeviceTypeKSec	&h39	One of the device type constants.
kDeviceTypeMailslot	&h0c	One of the device type constants.
kDeviceTypeMassStorage	&h2d	One of the device type constants.
kDeviceTypeMidiIn	&h0d	One of the device type constants.
kDeviceTypeMidiOut	&h0e	One of the device type constants.
kDeviceTypeModem	&h2b	One of the device type constants.
kDeviceTypeMouse	&h0f	One of the device type constants.
kDeviceTypeMultiUncProvider	&h10	One of the device type constants.
kDeviceTypeNamedPipe	&h11	One of the device type constants.
kDeviceTypeNetwork	&h12	One of the device type constants.
kDeviceTypeNetworkBrowser	&h13	One of the device type constants.
kDeviceTypeNetworkFileSystem	&h14	One of the device type constants.
kDeviceTypeNetworkRedirector	&h28	One of the device type constants.
kDeviceTypeNull	&h15	One of the device type constants.
kDeviceTypeParallelPort	&h16	One of the device type constants.
kDeviceTypePhysicalNetcard	&h17	One of the device type constants.

33.13 class WindowsDiscInfoMBS

33.13.1 class WindowsDiscInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to query details on a disc drive on Windows.
Example:

dim discs(-1) as WindowsDiscInfoMBS = WindowsDiscInfoMBS.Devices

for each disc as WindowsDiscInfoMBS in discs
MsgBox disc.ModelNumber
next

Blog Entries

- MBS Xojo Plugins, version 17.5pr8

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr4

33.13.2 Methods

33.13.3 Device(file as folderitem) as WindowsDiscInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the device with the given path.
Example:

// volume(0) is boot volume
dim disc as WindowsDiscInfoMBS = WindowsDiscInfoMBS.Device(volume(0))

Notes: Returns nil on any error (like missing permissions).
Seems like on Windows Vista/7 you need admin rights to query this.
The path of the folderitem must be something starting with a drive letter.
See also:

- 33.13.4 Device(path as string) as WindowsDiscInfoMBS 1060

33.13.4 Device(path as string) as WindowsDiscInfoMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the device with the given path.
Example:

dim disc as WindowsDiscInfoMBS = WindowsDiscInfoMBS.Device("C")

Notes: Returns nil on any error (like missing permissions).
Seems like on Windows Vista/7 you need admin rights to query this.
Path must be something starting with a drive letter.
See also:

33.13.5 Devices() as WindowsDiscInfoMBS()

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the device list.
Example:

// enumerate all drives
dim discs(-1) as WindowsDiscInfoMBS = WindowsDiscInfoMBS.Devices

Notes: Returns empty array on error.
May get more information on Windows Vista/7 if application runs as administrator.

33.13.6 Properties

33.13.7 BufferSize as Int64

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The buffer size of the disc controller.
Notes: This value is set if Mode contains 1.
(Read only property)

33.13.8 BytesPerSector as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of bytes per sector.
Notes: This value is set if Mode contains 4.
Typically 512 bytes, but that can increase in the future.
(Read and Write property)

33.13.9 Drive as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The drive number.
Notes: (Read and Write property)

33.13.10 Fixed as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this device is fixed.
Notes: This value is set if Mode contains 1 or 4.
(Read and Write property)

33.13.11 Mode as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This value shows which properties of the instance have been set with values.
Notes: Mode is a bitwise or combination of values 1, 2 or 4. So value is 6 if mode 2 and 4 are set. You can test with bitwiseand(mode,mask). For example if bitwiseAnd(mode,4)=4 then BytesPerSector is valid.
(Read and Write property)

33.13.12 ModelNumber as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The model number string for this device.
Example:

dim discs(-1) as WindowsDiscInfoMBS = WindowsDiscInfoMBS.Devices

for each disc as WindowsDiscInfoMBS in discs
MsgBox disc.ModelNumber
next

Notes: This string is not available for all discs.
This value is set if Mode contains 1 or 2.
(Read and Write property)

33.13.13   ProductRevision as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The product revision string.
Notes: This string is not available for all discs.
This value is set if Mode contains 2.
(Read and Write property)

33.13.14   Removable as Boolean

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this device is removable.
Notes: This value is set if Mode contains 1 or 4.
(Read and Write property)

33.13.15   RevisionNumber as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The revision number of the disc.
Notes: This string is not available for all discs.
This value is set if Mode contains 1.
(Read and Write property)

33.13.16   SectorsPerTrack as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of sectors per track.
Notes: This value is set if Mode contains 4.

(Read and Write property)

33.13.17   SerialNumber as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The serial number of the device.
Notes: This string is not available for all discs.
This value is set if Mode contains 1 or 2.
(Read and Write property)

33.13.18   Size as Int64

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of this disc in bytes.
Notes: This value is set if Mode contains 1 or 4.
(Read only property)

33.13.19   TracksPerCylinder as Integer

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of tracks per cylinder.
Notes: This value is set if Mode contains 4.
(Read and Write property)

33.13.20   VendorId as String

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The vendor ID of the disc.
Notes: This value is set if Mode contains 2.
This string is not available for all discs.
(Read and Write property)

33.14　class WindowsDisplayMBS

33.14.1　class WindowsDisplayMBS

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for details on displays.
Blog Entries

- MBS Xojo Plugins, version 23.2pr1

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.1

- MBS Xojo Plugins, version 18.1pr6

- MBS Xojo Plugins, version 18.1pr1

Videos

- Presentation from Munich conference about MBS Plugins.

- Presentation from Xojo Developer Conference 2019 in Miami.

33.14.2　Methods

33.14.3　Displays as WindowsDisplayMBS()

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries list of displays.
Notes: Returns nil in case of error.

33.14.4　Properties

33.14.5　DeviceInstanceID as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device instance ID in setup database.
Notes: (Read only property)

33.14.6　DeviceName as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device name of display.
Notes: (Read only property)

33.14.7   DisplayAdapterActive as Boolean

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether a monitor is presented as being "on" by the respective GDI view.
Notes: (Read only property)

33.14.8   DisplayAdapterDeviceID as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device ID for the display adapter.
Notes: (Read only property)

33.14.9   DisplayAdapterDeviceKey as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The display adapter device key.
Notes: (Read only property)

33.14.10   DisplayAdapterDeviceName as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The adapter device name.
Notes: (Read only property)

33.14.11   DisplayAdapterDeviceString as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device context string.
Notes: This is a description of the display adapter.
(Read only property)

33.14.12   DisplayAdapterRemovable as Boolean

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device is removable; it cannot be the primary display.
Notes: (Read only property)

33.14.13   DisplayAdapterStateFlags as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The state flags.
Notes: (Read only property)

33.14.14   DisplayMonitorActive as Boolean

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether a monitor is presented as being "on" by the respective GDI view.
Notes: (Read only property)

33.14.15   DisplayMonitorDeviceID as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device ID for the display monitor.
Notes: (Read only property)

33.14.16   DisplayMonitorDeviceKey as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The display monitor device key.
Notes: (Read only property)

33.14.17   DisplayMonitorDeviceName as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The monitor device name.
Notes: (Read only property)

33.14.18 DisplayMonitorDeviceString as String

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device context string.
Notes: This is a description of the display monitor.
(Read only property)

33.14.19 DisplayMonitorRemovable as Boolean

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device is removable; it cannot be the primary display.
Notes: (Read only property)

33.14.20 DisplayMonitorStateFlags as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The state flags.
Notes: (Read only property)

33.14.21 Height as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the display in pixels.
Notes: (Read only property)

33.14.22 HeightDPI as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DPI for the display.
Notes: Calculated based on HeightInch property.
Value is zero if we don't know.

(Read only property)

33.14.23   HeightInch as Double

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The physical height of the screen in inches.
Notes: Value is zero if we don't know.
(Read only property)

33.14.24   HeightMM as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The physical height of the screen in millimeters.
Notes: Value is zero if we don't know.
(Read only property)

33.14.25   LogPixelsX as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of pixels per logical inch along the screen width.
Notes: In a system with multiple display monitors, this value is the same for all monitors.

e.g. 200% is 96 * 2 = 192
(Read only property)

33.14.26   LogPixelsY as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of pixels per logical inch along the screen height.
Notes: In a system with multiple display monitors, this value is the same for all monitors.

e.g. 200% is 96 * 2 = 192
(Read only property)

33.14.27   MonitorHandle as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle for the monitor.
Notes: (Read only property)

33.14.28   MonitorHeight as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The monitor height.
Notes: (Read only property)

33.14.29   MonitorWidth as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The monitor width.
Notes: (Read only property)

33.14.30   MonitorX as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The position of monitor.
Notes: The display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read only property)

33.14.31   MonitorY as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The position of monitor.
Notes: The display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read only property)

33.14.32   Primary as Boolean

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this is primary display.
Notes: (Read only property)

33.14.33   Width as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the display in pixels.
Notes: (Read only property)

33.14.34   WidthDPI as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DPI for the display.
Notes: Calculated based on WidthInch property.
Value is zero if we don't know.
(Read only property)

33.14.35   WidthInch as Double

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The physical width of the screen in inches.
Notes: Value is zero if we don't know.
(Read only property)

33.14.36   WidthMM as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The physical width of the screen in millimeters.
Notes: Value is zero if we don't know.
(Read only property)

33.14.37   WorkHeight as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The work area height in pixels.
Notes: (Read only property)

33.14.38   WorkWidth as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The work area width in pixels.
Notes: (Read only property)

33.14.39   WorkX as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The work area position.
Notes: The work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area in rcMonitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read only property)

33.14.40   WorkY as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The work area position.
Notes: The work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area in rcMonitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read only property)

33.14.41   X as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The position of the display in the virtual desktop space.
Notes: Primary display is always at 0/0.

(Read only property)

33.14.42   Y as Integer

Plugin Version: 18.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The position of the display in the virtual desktop space.
Notes: Primary display is always at 0/0.
(Read only property)

33.15   class WindowsFileCopyMBS

33.15.1   class WindowsFileCopyMBS

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to help you copying files on Windows using system functions.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3

- MBS Xojo Plugins, version 19.6pr3

- MBS Xojo Plugins, version 18.1pr1

- MBS Xojo Plugins, version 17.6pr2

- Tip of day: Move file/folder to trash

- MBS Real Studio Plugins, version 12.4pr3

- Tip of the day: Move to trash

33.15.2   Methods

33.15.3   CopyFileEx(ExistingFileName as folderitem, NewFileName as folderitem, Flags as Integer) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies an existing file to a new file, notifying the application of its progress through the progress event.
Notes: Returns true on success and false on failure.

ExistingFileName: The name of an existing file.
NewFileName: The name of the new file.
Flags: Flags that specify how the file is to be copied. This parameter can be a combination of the CopyFile* constants.
See also:

- 33.15.4 CopyFileEx(ExistingFileName as String, NewFileName as String, Flags as Integer) as boolean 1073

33.15.4   CopyFileEx(ExistingFileName as String, NewFileName as String, Flags as Integer) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies an existing file to a new file, notifying the application of its progress through the progress event.
Notes: Returns true on success and false on failure.

ExistingFileName: The name of an existing file.
NewFileName: The name of the new file.
Flags: Flags that specify how the file is to be copied. This parameter can be a combination of the CopyFile* constants.
See also:

33.15.5 CopyFileSimple(ExistingFileName as folderitem, NewFileName as folderitem, FailIfExists as boolean=false) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies an existing file to a new file.
Notes: ExistingFileName: The name of an existing file.
NewFileName: The name of the new file.
FailIfExists: If this parameter is true and the new file specified by NewFileName already exists, the function fails. If this parameter is false and the new file already exists, the function overwrites the existing file and succeeds.

Returns true on success and false on failure.
See also:

33.15.6 CopyFileSimple(ExistingFileName as String, NewFileName as String, FailIfExists as boolean=false) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies an existing file to a new file.
Notes: ExistingFileName: The name of an existing file.
NewFileName: The name of the new file.
FailIfExists: If this parameter is true and the new file specified by NewFileName already exists, the function fails. If this parameter is false and the new file already exists, the function overwrites the existing file and succeeds.

Returns true on success and false on failure.
See also:

33.15.7 FileOperationCopy(source as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.8   FileOperationCopy(source as string, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.9   FileOperationCopy(source as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.

If a dialog is displayed, the window used in the parent property is used for the parent window.

If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.

See also:

33.15.10   FileOperationCopy(source() as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:

These names should be fully-qualified paths to prevent unexpected results.

Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position.

Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:

The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.

For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.

Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.

If a dialog is displayed, the window used in the parent property is used for the parent window.

If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.

See also:

33.15.11 FileOperationCopy(source() as folderitem, dest() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.

Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:

These names should be fully-qualified paths to prevent unexpected results.

Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.12   FileOperationCopy(source() as string, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

- 33.15.7 FileOperationCopy(source as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean   1075

- 33.15.8 FileOperationCopy(source as string, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean   1076

- 33.15.9 FileOperationCopy(source as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean   1077

- 33.15.10 FileOperationCopy(source() as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean   1078

- 33.15.11 FileOperationCopy(source() as folderitem, dest() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean   1079

- 33.15.13 FileOperationCopy(source() as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean   1082

- 33.15.14 FileOperationCopy(source() as string, dest() as string, Flags as Integer, ProgressTitle as string="") as boolean   1083

33.15.13 FileOperationCopy(source() as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.14   FileOperationCopy(source() as string, dest() as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copy the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.15   FileOperationDelete(file as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes files and folders.
Example:

dim w as new WindowsFileCopyMBS

dim f as FolderItem = SpecialFolder.Desktop.Child("test.test")

if w.FileOperationDelete(f, 0, "Hello") then
MsgBox "OK"
else
MsgBox "Failed "+str(w.Lasterror)
end if

Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.
See also:

- 33.15.18 FileOperationDelete(files() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean 1086

33.15.16 FileOperationDelete(filepathes as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes files and folders.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.
See also:

- 33.15.15 FileOperationDelete(file as folderitem, Flags as Integer, ProgressTitle as string="") as boolean 1084

- 33.15.17 FileOperationDelete(filepathes() as string, Flags as Integer, ProgressTitle as string="") as boolean 1085

- 33.15.18 FileOperationDelete(files() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean 1086

33.15.17 FileOperationDelete(filepathes() as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes files and folders.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.
See also:

- 33.15.15 FileOperationDelete(file as folderitem, Flags as Integer, ProgressTitle as string="") as boolean 1084

- 33.15.16 FileOperationDelete(filepathes as string, Flags as Integer, ProgressTitle as string="") as boolean 1085

33.15.18   FileOperationDelete(files() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes files and folders.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.
See also:

- 33.15.15 FileOperationDelete(file as folderitem, Flags as Integer, ProgressTitle as string="") as boolean 1084

- 33.15.16 FileOperationDelete(filepathes as string, Flags as Integer, ProgressTitle as string="") as boolean   1085

- 33.15.17 FileOperationDelete(filepathes() as string, Flags as Integer, ProgressTitle as string="") as boolean   1085

33.15.19   FileOperationMove(source as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.

For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.

Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.

If a dialog is displayed, the window used in the parent property is used for the parent window.

If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.

See also:

33.15.20 FileOperationMove(source as string, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.

Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.21 FileOperationMove(source as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.22 FileOperationMove(source() as folderitem, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.23   FileOperationMove(source() as folderitem, dest() as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position.
Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.24   FileOperationMove(source() as string, dest as folderitem, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.25   FileOperationMove(source() as string, dest as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.
Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:
These names should be fully-qualified paths to prevent unexpected results.
Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member

specifies FileOperationMultiDestFiles.

Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.

If a dialog is displayed, the window used in the parent property is used for the parent window.

If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.

See also:

33.15.26   FileOperationMove(source() as string, dest() as string, Flags as Integer, ProgressTitle as string="") as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Move the files specified in the source parameter to the location specified in the dest parameter.

Notes: This method has several varations so you can call it with a folderitem or a string to specify a file or folder. You can also pass an array of folderitems or strings to delete several files/folders in one operation.

source:

These names should be fully-qualified paths to prevent unexpected results.

Standard Microsoft MS-DOS wildcard characters, such as "*", are permitted only in the file-name position. Using a wildcard character elsewhere in the string will lead to unpredictable results.

dest:
The destination file or directory name. Wildcard characters are not allowed. Their use will lead to unpredictable results.

Copy and Move operations can specify destination directories that do not exist. In those cases, the system attempts to create them and normally displays a dialog box to ask the user if they want to create the new directory. To suppress this dialog box and have the directories created silently, set the FileOperationNoConfirmationMkDir flag in Flags.
For Copy and Move operations, the buffer can contain multiple destination file names if the fFlags member specifies FileOperationMultiDestFiles.
Use fully-qualified paths. Using relative paths is not prohibited, but can have unpredictable results.

Use the FileOperation* constants for the flags.
If a dialog is displayed, the window used in the parent property is used for the parent window.
If ProgressTitle is not empty, this string is used to specify the progress window title text.

Returns true on success and false on failure.
See also:

33.15.27 MoveFileSimple(ExistingFileName as folderitem, NewFileName as folderitem) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Moves an existing file or a directory, including its children.
Notes: ExistingFileName:The current name of the file or directory on the local computer.

NewFileName: The new name for the file or directory. The new name must not already exist. A new file may be on a different file system or drive. A new directory must be on the same drive.

Returns true on success and false on failure.

The MoveFile function will move (rename) either a file or a directory (including its children) either in the same directory or across directories. The one caveat is that the MoveFile function will fail on directory moves when the destination is on a different volume.

If a file is moved across volumes, MoveFile does not move the security descriptor with the file. The file will be assigned the default security descriptor in the destination directory.

The MoveFile function coordinates its operation with the link tracking service, so link sources can be tracked as they are moved.
See also:

33.15.28   MoveFileSimple(ExistingFileName as String, NewFileName as String) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Moves an existing file or a directory, including its children.
Notes: ExistingFileName:The current name of the file or directory on the local computer.

NewFileName: The new name for the file or directory. The new name must not already exist. A new file may be on a different file system or drive. A new directory must be on the same drive.

Returns true on success and false on failure.

The MoveFile function will move (rename) either a file or a directory (including its children) either in the same directory or across directories. The one caveat is that the MoveFile function will fail on directory moves when the destination is on a different volume.

If a file is moved across volumes, MoveFile does not move the security descriptor with the file. The file will be assigned the default security descriptor in the destination directory.

The MoveFile function coordinates its operation with the link tracking service, so link sources can be tracked as they are moved.
See also:

33.15.29 MoveFileWithProgress(ExistingFileName as folderitem, NewFileName as folderitem, Flags as Integer) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Moves a file or directory, including its children.
Notes: ExistingFileName: The name of the existing file or directory on the local computer.

If Flags specifies MOVEFILE_DELAY_UNTIL_REBOOT, the file cannot exist on a remote share because delayed operations are performed before the network is available.

NewFileName: The new name of the file or directory on the local computer.

When moving a file, NewFileName can be on a different file system or volume. If NewFileName is on another drive, you must set the MOVEFILE_COPY_ALLOWED flag in Flags.

When moving a directory, ExistingFileName and NewFileName must be on the same drive.

If Flags specifies MOVEFILE_DELAY_UNTIL_REBOOT and NewFileName is nil, MoveFileWithProgress registers ExistingFileName to be deleted when the system restarts. The function fails if it cannot access the registry to store the information about the delete operation. If ExistingFileName refers to a directory, the system removes the directory at restart only if the directory is empty.

Flags: The move options. This parameter can be one or more of the following values: MoveFileCopyAllowed, MoveFileReplaceExisting, MoveFileCreateHardLink, MoveFileDelayUntilReboot and MoveFileWriteThrough.

Return Value: If the function succeeds, the return value is true. If the function fails, the return value is false.

When moving a file across volumes, if the Progress event returns PROGRESS_CANCEL due to the user canceling the operation, MoveFileWithProgress will return zero and GetLastError will return ERROR_RE-QUEST_ABORTED. The existing file is left intact.

When moving a file across volumes, if the Progress event returns PROGRESS_STOP due to the user stopping the operation, MoveFileWithProgress will return zero and GetLastError will return ERROR_RE-QUEST_ABORTED. The existing file is left intact.
See also:

33.15.30 MoveFileWithProgress(ExistingFileName as String, NewFileName as String, Flags as Integer) as boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Moves a file or directory, including its children.
Notes: ExistingFileName: The name of the existing file or directory on the local computer.

If Flags specifies MOVEFILE_DELAY_UNTIL_REBOOT, the file cannot exist on a remote share because delayed operations are performed before the network is available.

NewFileName: The new name of the file or directory on the local computer.

When moving a file, NewFileName can be on a different file system or volume. If NewFileName is on another drive, you must set the MOVEFILE_COPY_ALLOWED flag in Flags.

When moving a directory, ExistingFileName and NewFileName must be on the same drive.

If Flags specifies MOVEFILE_DELAY_UNTIL_REBOOT and NewFileName is "", MoveFileWithProgress registers ExistingFileName to be deleted when the system restarts. The function fails if it cannot access the registry to store the information about the delete operation. If ExistingFileName refers to a directory, the system removes the directory at restart only if the directory is empty.

Flags: The move options. This parameter can be one or more of the following values: MoveFileCopyAllowed, MoveFileReplaceExisting, MoveFileCreateHardLink, MoveFileDelayUntilReboot and MoveFileWriteThrough.

Return Value: If the function succeeds, the return value is true. If the function fails, the return value is false.

When moving a file across volumes, if the Progress event returns PROGRESS_CANCEL due to the user canceling the operation, MoveFileWithProgress will return zero and GetLastError will return ERROR_RE-QUEST_ABORTED. The existing file is left intact.

When moving a file across volumes, if the Progress event returns PROGRESS_STOP due to the user stopping the operation, MoveFileWithProgress will return zero and GetLastError will return ERROR_RE-QUEST_ABORTED. The existing file is left intact.
See also:

33.15.31 Properties

33.15.32 Lasterror as Integer

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: This value is typically zero on success and -1 if the function is not supported by the plugin.

The FileOperation* functions have special error codes as following, but other methods have normal Windows Error codes:

DE_SAMEFILE	= &h71	The source and destination files are the same file.
DE_MANYSRC1DEST	= &h72	Multiple file paths were specified in the source buffer, but only one destination file path.
DE_DIFFDIR	= &h73	Rename operation was specified but the destination path is a different directory. Use the move operation instead.
DE_ROOTDIR	= &h74	The source is a root directory, which cannot be moved or renamed.
DE_OPCANCELLED	= &h75	The operation was cancelled by the user, or silently cancelled if the appropriate flags were supplied to SHFileOperation.
DE_DESTSUBTREE	= &h76	The destination is a subtree of the source.
DE_ACCESSDENIEDSRC	= &h78	Security settings denied access to the source.
DE_PATHTOODEEP	= &h79	The source or destination path exceeded or would exceed MAX_PATH.
DE_MANYDEST	= &h7A	The operation involved multiple destination paths, which can fail in the case of a move operation.
DE_INVALIDFILES	= &h7C	The path in the source or destination or both was invalid.
DE_DESTSAMETREE	= &h7D	The source and destination have the same parent folder.
DE_FLDDESTISFILE	= &h7E	The destination path is an existing file.
DE_FILEDESTISFLD	= &h80	The destination path is an existing folder.
DE_FILENAMETOOLONG	= &h81	The name of the file exceeds MAX_PATH.
DE_DEST_IS_CDROM	= &h82	The destination is a read-only CD-ROM, possibly unformatted.
DE_DEST_IS_DVD	= &h83	The destination is a read-only DVD, possibly unformatted.
DE_DEST_IS_CDRECORD	= &h84	The destination is a writable CD-ROM, possibly unformatted.
DE_FILE_TOO_LARGE	= &h85	The file involved in the operation is too large for the destination media or file system.
DE_SRC_IS_CDROM	= &h86	The source is a read-only CD-ROM, possibly unformatted.
DE_SRC_IS_DVD	= &h87	The source is a read-only DVD, possibly unformatted.
DE_SRC_IS_CDRECORD	= &h88	The source is a writable CD-ROM, possibly unformatted.
DE_ERROR_MAX	= &hB7	MAX_PATH was exceeded during the operation.
UnknownError	= &h402	An unknown error occurred. This is typically due to an invalid path in the source or destination. This error does not occur on Windows Vista and later.
ERRORONDEST	= &h10000	An unspecified error occurred on the destination.
DE_ROOTDIR | ERRORONDEST	= &h10074	Destination is a root directory and cannot be renamed.

(Read and Write property)

33.15.33 MultiThreaded as Boolean

Plugin Version: 12.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to use helper thread for better GUI responsiveness.
Notes: This property allows you to call this method in a Thread. The plugin will perform the actual operation on a helper thread, so it does not block Xojo's threading. Your GUI can continue to run this way. Progress event fires asynchronously.
(Read and Write property)

33.15.34   OperationsAborted as Boolean

Plugin Version: 9.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: This flag is set by the FileOperation method to indicate whether the operation was aborted.
Notes: (Read and Write property)

33.15.35   Parent as Variant

Plugin Version: 9.2, Platform: Windows, Targets: Desktop only.

Function: The parent window for the FileOperation methods.
Notes: Can reference a Window or DesktopWindow object.
(Read and Write property)

33.15.36   Events

33.15.37   Progress(TotalFileSize as int64, TotalBytesTransferred as int64, StreamSize as int64, StreamBytesTransferred as Int64, StreamNumber as Integer, Reason as Integer) as Integer

Plugin Version: 9.2, Platform: Windows, Targets: .

Function: It is called when a portion of a copy or move operation is completed.
Notes: This event is called by the CopyFileEx and the MoveFileWithProgress methods.

TotalFileSize: The total size of the file, in bytes.
TotalBytesTransferred: The total number of bytes transferred from the source file to the destination file since the copy operation began.
StreamSize: The total size of the current file stream, in bytes.
StreamBytesTransferred: The total number of bytes in the current stream that have been transferred from the source file to the destination file since the copy operation began.
StreamNumber: A handle to the current stream. The first time CopyProgressRoutine is called, the stream number is 1.
Reason: CallbackChunkFinished or CallbackStreamSwitched.

Return ProgressCancel, ProgressStop, ProgressContinue or ProgressQuiet.

If you use multithreading option, you may receive additional events after you cancelled or stopped the operation due to the events coming asynchronously.

33.15.38 Constants

Constants

Constant	Value	Description
CallbackChunkFinished	0	One of the reason constants for the Progress event. Another part of the data file was copied.
CallbackStreamSwitched	1	One of the reason constants for the Progress event. Another stream was created and is about to be copied. This is th reason given when the callback routine is first invoked.
CopyFileAllowDecryptedDestination	8	One of the constants used with CopyFileEx. An attempt to copy an encrypted file will succeed even if the destina cannot be encrypted. Windows 2000: This value is not supported.
CopyFileCopySymLink	&h800	One of the constants used with CopyFileEx. If the source file is a symbolic link, the destination file is also a syn pointing to the same file that the source symbolic link is pointing to. Server 2003 and Windows XP/2000: This value is not supported.
CopyFileFailIfExists	1	One of the constants used with CopyFileEx. The copy operation fails immediately if the target file already exists
CopyFileOpenSourceForWrite	4	One of the constants used with CopyFileEx. The file is copied and the original file is opened for write access.
CopyFileRestartable	2	One of the constants used with CopyFileEx. Progress of the copy is tracked in the target file in case the copy failed copy can be restarted at a later time by specifying the same source and dest as those used in the call that failed.
FileOperationAllowUndo	&h40	One of the flag constants for the FileOperation methods. Preserve undo information, if possible. Prior to Windows Vista, operations could be undone only from the sa that performed the original operation. In Windows Vista and later systems, the scope of the undo is a us Any process running in the user session can undo another operation. state is held in the Explorer.exe process, and as long as that process i it can coordinate the undo functions. If the source file parameter does not contain fully qualified path and this flag is ignored.
FileOperationFilesOnly	&h80	One of the flag constants for the FileOperation methods. Perform the operation only on files (not on folders) if a wildcard file r is specified.
FileOperationMultiDestFiles	1	One of the flag constants for the FileOperation methods. The destination multiple destination files (one for each source file rather than one directory where all source files are to be deposited.
FileOperationNoConfirmation	&h10	One of the flag constants for the FileOperation methods. Respond with Yes to All for any dialog box that is displayed.
FileOperationNoConfirmationMkDir	&h200	One of the flag constants for the FileOperation methods. Do not ask the user to confirm the creation of a new directory if the requires one to be created.
FileOperationNoConnectedElements	&H2000	One of the flag constants for the FileOperation methods. Do not move connected files as a group. Only move the specified fil
FileOperationNoCopySecurityAttributes	&h800	One of the flag constants for the FileOperation methods. Do not copy the security attributes of the file. The destination file re security attributes of its new folder.
FileOperationNoErrorUI	&h400	One of the flag constants for the FileOperation methods. Do not display a dialog to the user if an error occurs.
FileOperationNoRecursion	&h1000	One of the flag constants for the FileOperation methods. Only perform the operation in the local directory. Don't operate r into subdirectories, which is the default behavior.
FileOperationRenameCollision	8	One of the flag constants for the FileOperation methods. Give the file being operated on a new name in a move, copy, or rename if a file with the target name already exists at the destination.
FileOperationSilent	4	One of the flag constants for the FileOperation methods. Do not display a progress dialog box.
FileOperationSimpleProgress	&h100	One of the flag constants for the FileOperation methods. Display a progress dialog box but do not show individual file nam

33.16 class WindowsFileInfoMBS

33.16.1 class WindowsFileInfoMBS

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The window file information class.
Example:

dim f as FolderItem = SelectFolder

if f<>Nil then
dim w as new WindowsFileInfoMBS(f)
MsgBox str(w.FileIndex)
end if

Notes: The identifier that is stored in the FileIndex members is called the file ID. Support for file IDs is file system-specific. File IDs are not guaranteed to be unique over time, because file systems are free to reuse them. In some cases, the file ID for a file can change over time.

In the FAT file system, the file ID is generated from the first cluster of the containing directory and the byte offset within the directory of the entry for the file. Some defragmentation products change this byte offset. (Windows in-box defragmentation does not.) Thus, a FAT file ID can change over time. Renaming a file in the FAT file system can also change the file ID, but only if the new file name is longer than the old one.

In the NTFS file system, a file keeps the same file ID until it is deleted. You can replace one file with another file without changing the file ID by using the ReplaceFile function. However, the file ID of the replacement file, not the replaced file, is retained as the file ID of the resulting file.

Not all file systems can record creation and last access time, and not all file systems record them in the same manner. For example, on a Windows FAT file system, create time has a resolution of 10 milliseconds, write time has a resolution of 2 seconds, and access time has a resolution of 1 day (the access date). On the NTFS file system, access time has a resolution of 1 hour. For more information, see File Times.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.5

- MBS Xojo Plugins, version 18.5pr8

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

33.16.2 Methods

33.16.3 Constructor(file as folderitem)

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor using a folderitem.
Notes: On success the valid property is set to true.
See also:

33.16.4 Constructor(handle as Integer)

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor using a Windows File Handle.
Notes: On success the valid property is set to true.
See also:

33.16.5 Constructor(path as string)

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor using a file path.
Notes: On success the valid property is set to true.
See also:

33.16.6 Constructor(stream as BinaryStream)

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor using a binarystream.
Notes: On success the valid property is set to true.
See also:

33.16.7 Properties

33.16.8 CreationDate as Date

Plugin Version: 18.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: A date that specifies when a file or directory is created.
Notes: If the underlying file system does not support creation time, this member is nil.
(Read only property)

33.16.9 CreationTime as UInt64

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: An UInt64 that specifies when a file or directory is created.
Notes: If the underlying file system does not support creation time, this member is zero (0).
(Read only property)

33.16.10 FileAttributes as Integer

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file attributes.
Notes: For possible values and their descriptions, see kFileAttribute* constants.
(Read only property)

33.16.11 FileIndex as UInt64

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The unique identifier that is associated with a file.
Notes: The identifier and the volume serial number uniquely identify a file on a single computer. To determine whether two open handles represent the same file, combine the identifier and the volume serial number for each file and compare them.
(Read only property)

33.16.12   FileSize as UInt64

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file size.
Notes: (Read only property)

33.16.13   LastAccessDate as Date

Plugin Version: 18.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: For a file, the structure specifies the last time that a file is read from or written to.
Notes: For a directory, the structure specifies when the directory is created. For both files and directories, the specified date is correct, but the time of day is always set to midnight. If the underlying file system does not support the last access time, this member is nil.
(Read only property)

33.16.14   LastAccessTime as UInt64

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For a file, the structure specifies the last time that a file is read from or written to.
Notes: For a directory, the structure specifies when the directory is created. For both files and directories, the specified date is correct, but the time of day is always set to midnight. If the underlying file system does not support the last access time, this member is zero (0).
(Read only property)

33.16.15   LastWriteDate as Date

Plugin Version: 18.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: For a file, the structure specifies the last time that a file is written to.
Notes: For a directory, the structure specifies when the directory is created. If the underlying file system does not support the last write time, this member is nil.

(Read only property)

33.16.16 LastWriteTime as UInt64

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: For a file, the structure specifies the last time that a file is written to.
Notes: For a directory, the structure specifies when the directory is created. If the underlying file system does not support the last write time, this member is zero (0).
(Read only property)

33.16.17 NumberOfLinks as Integer

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of links to this file.
Notes: For the FAT file system this member is always 1. For the NTFS file system, it can be more than 1.
(Read only property)

33.16.18 Valid as Boolean

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the constructor got the values.
Notes: (Read only property)

33.16.19 VolumeSerialNumber as Integer

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The serial number of the volume that contains a file.
Notes: (Read only property)

33.16.20 Constants

Constants

Constant	Value	Description
kFileAttributeArchive	32	One of the file attribute constants. A file or directory that is an archive file or directory. Applications typicall use this attribute to mark files for backup or removal.
kFileAttributeCompressed	2048	One of the file attribute constants. A file or directory that is compressed. For a file, all of the data in the file compressed. For a directory, compression is the default for newly created file and subdirectories.
kFileAttributeDevice	64	One of the file attribute constants. This value is reserved for system use.
kFileAttributeDirectory	16	One of the file attribute constants. The handle that identifies a directory.
kFileAttributeEncrypted	16384	One of the file attribute constants. A file or directory that is encrypted. For a file, all data streams in the file ar encrypted. For a directory, encryption is the default for newly created files an subdirectories.
kFileAttributeHidden	2	One of the file attribute constants. The file or directory is hidden. It is not included in an ordinary director listing.
kFileAttributeNormal	128	One of the file attribute constants. A file that does not have other attributes set. This attribute is valid only whe used alone.
kFileAttributeNotContentIndexed	8192	One of the file attribute constants. The file or directory is not to be indexed by the content indexing service.
kFileAttributeOffline	4096	One of the file attribute constants. The data of a file is not available immediately. This attribute indicates that th file data is physically moved to offline storage. This attribute is used by Remot Storage, which is the hierarchical storage management software. Application should not arbitrarily change this attribute.
kFileAttributeReadonly	1	One of the file attribute constants. A file that is read-only. Applications can read the file, but cannot write to it delete it. This attribute is not honored on directories. For more information see "You cannot view or change the Read-only or the System attributes folders in Windows Server 2003, in Windows XP, or in Windows Vista".
kFileAttributeReparsePoint	1024	One of the file attribute constants. A file or directory that has an associated reparse point, or a file that is symbolic link.
kFileAttributeSparseFile	512	One of the file attribute constants. A file that is a sparse file.
kFileAttributeSystem	4	One of the file attribute constants. A file or directory that the operating system uses a part of, or uses exclusivel
kFileAttributeTemporary	256	One of the file attribute constants. A file that is being used for temporary storage. File systems avoid writing dat back to mass storage if sufficient cache memory is available, because typicall an application deletes a temporary file after the handle is closed. In tha scenario, the system can entirely avoid writing the data. Otherwise, the dat is written after the handle is closed.
kFileAttributeVirtual	65536	One of the file attribute constants. This value is reserved for system use.

33.17 class WindowsFileStreamMBS

33.17.1 class WindowsFileStreamMBS

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for file streams on Windows.
Notes: Windows can store more than one data stream in a file.

(on Mac/Linux, please check the ExtendedAttributesMBS module)
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 16.4pr5

33.17.2 Methods

33.17.3 Constructor

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The private constructor.

33.17.4 List(file as folderitem) as WindowsFileStreamMBS()

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Lists file streams in a file at given folderitem.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("test.txt")
dim w() as WindowsFileStreamMBS = WindowsFileStreamMBS.List(f)

Break

Notes: Returns nil in case of error.
Else returns an array with stream details.

Requires Windows Vista or newer. Raises exception on Windows XP.
See also:

- 33.17.5 List(Path as String) as WindowsFileStreamMBS() 1110

33.17.5   List(Path as String) as WindowsFileStreamMBS()

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Lists file streams in a file at given path.
Notes: Returns nil in case of error.
Else returns an array with stream details.

Requires Windows Vista or newer. Raises exception on Windows XP.
See also:

33.17.6   Properties

33.17.7   Name as String

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the stream.
Notes: (Read only property)

33.17.8   Size as UInt64

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size in bytes.
Notes: (Read only property)

33.18 class WindowsFileVersionMBS

33.18.1 class WindowsFileVersionMBS

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to read Windows version information from a file.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetInternalName
end if

Blog Entries

- MBS Plugins 10.3 Release Notes

- Tipp of the day: WindowsFileVersionMBS

33.18.2 Methods

33.18.3 FileVersion as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file version in an user readable string.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.FileVersion
end if

Notes: format is "1.2.3.4".

33.18.4 GetCompanyName as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key CompanyName.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetCompanyName
end if
```

33.18.5   GetFileDescription as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key FileDescription.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetFileDescription
end if
```

33.18.6   GetFileVersion as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key FileVersion.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetFileVersion
end if
```

33.18.7   GetInternalName as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key InternalName.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetInternalName
end if

33.18.8   GetLegalCopyright as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key LegalCopyright.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetLegalCopyright
end if

33.18.9   GetOriginalFilename as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key OriginalFilename.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetOriginalFilename
end if

33.18.10   GetProductName as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key ProductName.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS

if w.OpenFile(f) then
MsgBox w.GetProductName
end if

33.18.11   GetProductVersion as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns value for key ProductVersion.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetProductVersion
end if

33.18.12   OpenFile(file as folderitem) as boolean

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Loads file information from a given file.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox w.GetCompanyName
end if

Notes: Returns true on success and false on failure.

33.18.13   ProductVersion as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The product version in an user readable string.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS

```
if w.OpenFile(f) then
MsgBox w.ProductVersion
end if
```

Notes: format is "1.2.3.4"

33.18.14   QueryBinaryValue(key as string) as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Low level function to read a binary item from the information resource.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
// we define encoding as UTF16 as we know the value for ProductName is an UTF16 string.
MsgBox DefineEncoding(w.QueryBinaryValue("ProductName"), encodings.UTF16)
end if
```

Notes: Returns "" on any error.

33.18.15   QueryUnicodeValue(key as string) as string

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Low level function to read an unicode string item from the information resource.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.QueryUnicodeValue("ProductName"))
end if
```

Notes: Returns "" on any error.

33.18.16 Properties

33.18.17 FileDateLS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: File date number. Lower part.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileDateLS)
end if
```

Notes: Filedate is a 64 bit integer. This is the lower 32 bit part.

Specifies the least significant 32 bits of the file's 64-bit binary creation date and time stamp.
(Read and Write property)

33.18.18 FileDateMS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: File date number. Higher part.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileDateMS)
end if
```

Notes: Filedate is a 64 bit integer. This is the higher 32 bit part.

Specifies the most significant 32 bits of the file's 64-bit binary creation date and time stamp.
(Read and Write property)

33.18.19 FileFlags as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file flags.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileFlags)
end if

Notes: Contains a bitmask that specifies the Boolean attributes of the file. This member can include one or more of the following values.

VS_FF_DEBUG	= 1	The file contains debugging information or is compiled with debugging features enabled.
VS_FF_INFOINFERRED	= 16	The file's version structure was created dynamically; therefore, some of the members in this structure may be empty or incorrect. This flag should never be set in a file's VS_VERSIONINFO data.
VS_FF_PATCHED	= 4	The file has been modified and is not identical to the original shipping file of the same version number.
VS_FF_PRERELEASE	= 2	The file is a development version, not a commercially released product.
VS_FF_PRIVATEBUILD	= 8	The file was not built using standard release procedures. If this flag is set, the StringFileInfo structure should contain a PrivateBuild entry.
VS_FF_SPECIALBUILD	= 32	The file was built by the original company using standard release procedures but is a variation of the normal file of the same version number. If this flag is set, the StringFileInfo structure should contain a SpecialBuild entry.

(Read and Write property)

33.18.20 FileOS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file os settings.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileOS)
end if

Notes: Specifies the operating system for which this file was designed. This member can be one of the following values.

VOS_DOS	= &h00010000	The file was designed for MS-DOS.
VOS_NT	= &h00040000	The file was designed for Windows NT.
VOS___WINDOWS16	= &h00000001	The file was designed for 16-bit Windows.
VOS___WINDOWS32	= &h00000004	The file was designed for 32-bit Windows.
VOS_OS216	= &h00020000	The file was designed for 16-bit OS/2.
VOS_OS232	= &h00030000	The file was designed for 32-bit OS/2.
VOS___PM16	= &h00000002	The file was designed for 16-bit Presentation Manager.
VOS___PM32	= &h00000003	The file was designed for 32-bit Presentation Manager.
VOS_UNKNOWN	= &h00000000	The operating system for which the file was designed is unknown to the system.

An application can combine these values to indicate that the file was designed for one operating system running on another. The following FileOS values are examples of this, but are not a complete list.

VOS_DOS_WINDOWS16	= &h00010001	The file was designed for 16-bit Windows running on MS-DOS.
VOS_DOS_WINDOWS32	= &h00010004	The file was designed for 32-bit Windows running on MS-DOS.
VOS_NT_WINDOWS16	= ?	The file was designed for Windows NT.
VOS_OS216_PM16	= &h00020002	The file was designed for 16-bit Presentation Manager running on 16-bit OS/2.
VOS_OS232_PM32	= &h00030003	The file was designed for 32-bit Presentation Manager running on 32-bit OS/2.

(Read and Write property)

33.18.21   FileSubtype as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Windows file subtype code.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileSubType)
end if
```

Notes: Specifies the function of the file. The possible values depend on the value of FileType. For all values of FileType not described in the following list, FileSubtype is zero.

If FileType is VFT_DRV, FileSubtype can be one of the following values.

If FileType is VFT_FONT, FileSubtype can be one of the following values.

If FileType is VFT_VXD, FileSubtype contains the virtual device identifier included in the virtual device

VFT2_UNKNOWN	= 0	The driver type is unknown by the system.
VFT2_DRV_COMM	= 10	The file contains a communications driver.
VFT2_DRV_PRINTER	= 1	The file contains a printer driver.
VFT2_DRV_KEYBOARD	= 2	The file contains a keyboard driver.
VFT2_DRV_LANGUAGE	= 3	The file contains a language driver.
VFT2_DRV_DISPLAY	= 4	The file contains a display driver.
VFT2_DRV_MOUSE	= 5	The file contains a mouse driver.
VFT2_DRV_NETWORK	= 6	The file contains a network driver.
VFT2_DRV_SYSTEM	= 7	The file contains a system driver.
VFT2_DRV_INSTALLABLE	= 8	The file contains an installable driver.
VFT2_DRV_SOUND	= 9	The file contains a sound driver.
VFT2_DRV_VERSIONED_PRINTER	= 10	The file contains a versioned printer driver.

VFT2_UNKNOWN	= 0	The font type is unknown by the system.
VFT2_FONT_RASTER	= 1	The file contains a raster font.
VFT2_FONT_VECTOR	= 2	The file contains a vector font.
VFT2_FONT_TRUETYPE	= 3	The file contains a TrueType font.

control block.
All FileSubtype values not listed here are reserved.
(Read and Write property)

33.18.22   FileType as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Windows file type code.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileType)
end if

Notes: Specifies the general type of file. This member can be one of the following values. All other values are reserved.

(Read and Write property)

33.18.23   FileVersionLS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

VFT_UNKNOWN	= 0	The file type is unknown to the system.
VFT_APP	= 1	The file contains an application.
VFT_DLL	= 2	The file contains a DLL.
VFT_DRV	= 3	The file contains a device driver. If dwFileType is VFT_DRV, FileSubtype contains a more specific description of the driver.
VFT_FONT	= 4	The file contains a font. If dwFileType is VFT_FONT, FileSubtype contains a more specific description of the font file.
VFT_VXD	= 5	The file contains a virtual device.
VFT_STATIC_LIB	= 7	The file contains a static-link library.

Function: File version number. Lower part.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileVersionLS)
end if

Notes: Fileversion is a 64 bit integer. This is the lower 32 bit part.

Specifies the least significant 32 bits of the file's binary version number. This member is used with FileVersionMS to form a 64-bit value used for numeric comparisons.
(Read and Write property)

33.18.24   FileVersionMS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: File version number. Higher part.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.FileVersionMS)
end if

Notes: Fileversion is a 64 bit integer. This is the higher 32 bit part.

Specifies the most significant 32 bits of the file's binary version number. This member is used with FileVer-

sionLS to form a 64-bit value used for numeric comparisons.
(Read and Write property)

33.18.25   LangCharset as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The language charset to use.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.LangCharset)
end if

Notes: OpenFile sets this to the default value.
You can change it later.
Value is used on each call of GetOriginalFilename, GetFileVersion, GetCompanyName, GetProductName, GetInternalName, GetProductVersion, GetFileDescription, GetLegalCopyright, QueryUnicodeValue and QueryBinaryValue.
(Read and Write property)

33.18.26   Lasterror as Integer

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: Lasterror means no error has happend.
Normally a Windows error code.
(Read and Write property)

33.18.27   ProductVersionLS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Product version number. Lower part.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then

MsgBox str(w.ProductVersionLS)
end if

Notes: Productversion is a 64 bit integer. This is the lower 32 bit part.

Specifies the least significant 32 bits of the binary version number of the product with which this file was distributed. This member is used with ProductVersionMS to form a 64-bit value used for numeric comparisons.
(Read and Write property)

33.18.28   ProductVersionMS as Integer

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Product version number. Higher part.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.ProductVersionMS)
end if

Notes: Productversion is a 64 bit integer. This is the higher 32 bit part.

Specifies the most significant 32 bits of the binary version number of the product with which this file was distributed. This member is used with ProductVersionLS to form a 64-bit value used for numeric comparisons.
(Read and Write property)

33.18.29   RawData as String

Plugin Version: 10.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the data of the file version record as string.
Notes: (Read only property)

33.18.30 Success as Boolean

Plugin Version: 6.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether last call to OpenFile was successful.
Example:

dim f as FolderItem = SpecialFolder.Desktop.Child("dbgview.exe")
dim w as new WindowsFileVersionMBS
if w.OpenFile(f) then
MsgBox str(w.Success)
end if

Notes: (Read and Write property)

33.19　class WindowsGraphicsDeviceContextMBS

33.19.1　class WindowsGraphicsDeviceContextMBS

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: The class to safely access a HDC of a graphics object.
Notes: The plugin will get the handle in constructor and release it in destructor/close.
Blog Entries

- MBS Xojo Plugins, version 21.1pr4

- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.3

- MBS Xojo Plugins, version 20.3pr5

33.19.2　Methods

33.19.3　Close

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: Closes the handle by releasing memory.
Notes: Same as destructor.

33.19.4　Constructor(c as control)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop only.

Function: The constructor for a control object.
Notes: Queries HDC for control object.
Raises exceptions on failure.
See also:

33.19.5 Constructor(c as DesktopControl)

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

Function: The constructor for a control object.
Notes: Queries HDC for control object.
Raises exceptions on failure.
See also:

33.19.6 Constructor(g as graphics)

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: The constructor for a graphics object.
Notes: Queries HDC for graphics object.
Raises exceptions on failure.
See also:

33.19.7 Constructor(p as Picture)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop only.

Function: The constructor for a picture object.
Notes: Queries HDC for picture object.
Raises exceptions on failure.
See also:

33.19.8   Constructor(w as DesktopWindow)

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

Function: The constructor for a window object.
Notes: Queries HDC for window object.
Raises exceptions on failure.
See also:

33.19.9   Constructor(w as window)

Plugin Version: 21.1, Platform: Windows, Targets: Desktop only.

Function: The constructor for a window object.
Notes: Queries HDC for window object.
Raises exceptions on failure.
See also:

33.19.10   Properties

33.19.11   Graphics as Graphics

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: The references graphics object.
Notes: (Read only property)

33.19.12   Handle as Integer

Plugin Version: 20.3, Platform: Windows, Targets: Desktop only.

Function: The handle.
Notes: This is HDC to be used with declares.
(Read only property)

33.20   class WindowsGUIResourcesMBS

33.20.1   class WindowsGUIResourcesMBS

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Class to retrieves the count of handles to graphical user interface (GUI) objects in use by the specified process.
Notes: Allows you to monitor how many handles you use from GDI.
As they are limited, your app will crash if it hits a limit.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 18.2

- Windows GUI Resources

- MBS Xojo Plugins, version 18.2pr8

Videos

- Presentation from Munich conference about MBS Plugins.

- Presentation from Xojo Developer Conference 2019 in Miami.

33.20.2   Methods

33.20.3   Constructor

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Examines current process and sets properties.

If the constructor succeeds, the properties are set to the count of handles to GUI objects in use by the process. If no GUI objects are in use, the return value is zero.

A process without a graphical user interface does not use GUI resources, therefore, GetGuiResources will return zero.
See also:

- 33.20.4 Constructor(ProcessID as integer)   1128

33.20.4   Constructor(ProcessID as integer)

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: Examines given process and sets properties.

If the constructor succeeds, the properties are set to the count of handles to GUI objects in use by the process. If no GUI objects are in use, the return value is zero.

A process without a graphical user interface does not use GUI resources, therefore, GetGuiResources will return zero.
See also:

33.20.5   Properties

33.20.6   GDIObjectCount as Integer

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The count of GDI objects.
Notes: (Read only property)

33.20.7   GDIObjectPeak as Integer

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak count of GDI objects.
Notes: (Read only property)

33.20.8   LastError as Integer

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error from Windows.
Notes: Should be zero on success.
But could give error code like permissions denied, if you want to examine another process.
(Read only property)

33.20.9   UserObjectCount as Integer

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The count of USER objects.
Notes: (Read only property)

33.20.10   UserObjectPeak as Integer

Plugin Version: 18.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The peak count of USER objects.
Notes: (Read only property)

33.21 class WindowsIniMBS

33.21.1 class WindowsIniMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to handle INI files on Windows.
Notes: Ini files are build by this:

[Sectionname]
key1=value1

[Appname]
key2=value2

An ini file contains one or more sections which are labeled with a name of a section or in the Win.ini file with the application name.
Each section is filled with one or more key = value pairs.
Blog Entries

- MBS Real Studio Plugins, version 13.0pr10

33.21.2 Methods

33.21.3 GetPrivateProfileInt(appname as string, keyname as string, default-Value as Integer = 0) as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads an integer value from the ini file.
Notes: This is the function to work on a private ini file.
Returns the default value on any error.

33.21.4 GetPrivateProfileSection(appname as string) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns a section from an ini file.
Notes: This is the function to work on a private ini file.
The string is built using CStrings (with ending chr(0)).
Returns ”” on any error.

33.21.5 GetPrivateProfileString(appname as string, keyname as string, default-Value as string = "") as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads a string from an ini file.
Example:

dim l as new WindowsIniMBS

l.filename="c:\test.ini"

dim value as string = l.GetPrivateProfileString("appname","keyname")
msgBox "Read: "+value

Notes: This is the function to work on a private ini file.
Returns the default value on any error.

33.21.6 GetPrivateProfileStruct(section as string, keyname as string, size as Integer) as memoryblock

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads a structure from an ini file.
Notes: This is the function to work on a private ini file.
Size is the size of the memoryblock returned.
Returns nil on any error.
A checksum is checked to verify that the data is unchanged.

33.21.7 GetProfileInt(appname as string, keyname as string, defaultValue as Integer = 0) as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an integer value of the Win.ini file.
Notes: Returns the default value on any error.

33.21.8 GetProfileSection(appname as string) as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns a profile section of the Win.ini file.
Notes: The string is built using CStrings (with ending chr(0)).

Returns "" on any error.

33.21.9 GetProfileString(appname as string, keyname as string, defaultValue as string = "") as string

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns an integer value of the Win.ini file.
Example:

dim l as new WindowsIniMBS

l.filename="c:\test.ini"

dim value as string = l.GetProfileString("appname","keyname")
msgBox "Read: "+value

Notes: Searches for a string.
All three parameters are not allowed to be empty. And you must pass trimmed strings (no space on the left and/or right side)
Returns the default string on any error.

33.21.10 WritePrivateProfileSection(appname as string, value as string) as boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Write a section to an ini file.
Notes: This is the function to work on a private ini file.
Value is an array of strings concated with chr(0) as the delimiter.
Returns true if successfull.

33.21.11 WritePrivateProfileString(appname as string, keyname as string, value as string) as boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes a string to an ini file.
Example:

dim l as windowsIniMBS

l=new windowsIniMBS
l.filename="c:\test.ini"

if l.writePrivateProfileString("appname","keyname","value") then
msgBox "ok"
end if

Notes: This is the function to work on a private ini file.

33.21.12   WritePrivateProfileStruct(section as string, keyname as string, mem as memoryblock, size as Integer) as boolean

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes a memoryblock as a structure to a ini file.
Notes: This is the function to work on a private ini file.
Returns true if successfull.
A checksum is stored to verify later whether the data was changed.

33.21.13   Properties

33.21.14   BufferSize as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the buffer used for reading strings from the INI files.
Notes: Should not be bigger than 32767 bytes on Windows 95/98/ME.
Default size is 32000.
(Read and Write property)

33.21.15   Filename as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The file path of the private profile file to use.
Notes: e.g. "C:\Test.ini" or just "myapp.ini".
(Read and Write property)

33.22 class WindowsKeyboardLayoutMBS

33.22.1 class WindowsKeyboardLayoutMBS

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a keyboard layout on Windows.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr8

33.22.2 Methods

33.22.3 Constructor

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new keyboard layout object with current layout.
Example:

dim w as new WindowsKeyboardLayoutMBS
MsgBox w.name

See also:

33.22.4 Constructor(SubLanguageID as Integer, PrimaryLanguageID as Integer)

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new keyboard layout object with the given ID.
See also:

33.22.5 KeyboardLayoutName as string

Plugin Version: 12.5, Platform: Windows, Targets: Desktop only.

Function: Returns the name of the current keyboard layout name.
Example:

msgbox WindowsKeyboardLayoutMBS.KeyboardLayoutName

33.22.6 List as WindowsKeyboardLayoutMBS()

Plugin Version: 12.5, Platform: Windows, Targets: Desktop only.

Function: Returns a list of all keyboard layouts installed.
Example:

dim list() as WindowsKeyboardLayoutMBS = WindowsKeyboardLayoutMBS.List
MsgBox str(UBound(list)+1)+" keyboard layouts."

33.22.7 Properties

33.22.8 Handle as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal ID of the keyboard layout.
Notes: (Read only property)

33.22.9 Name as String

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the name of the keyboard layout.
Example:

dim w as new WindowsKeyboardLayoutMBS
MsgBox w.name

Notes: (Read only property)

33.22.10 PrimaryLanguageID as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The primary ID for this language.

Notes: (Read and Write property)

33.22.11 SubLanguageID as Integer

Plugin Version: 12.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The sub language ID.
Notes: This is needed as there are for example 16 different English variants.
(Read and Write property)

33.22.12 Constants

Language Constants.

Constant	Value	Description
LangAfrikaans	&h36	
LangAlbanian	&h1C	
LangAlsatian	&h84	
LangAmharic	&h5E	
LangArabic	&h01	
LangArmenian	&h2B	
LangAssamese	&h4D	
LangAzeri	&h2C	
LangBashkir	&h6D	
LangBasque	&h2D	
LangBelarusian	&h23	
LangBengali	&h45	
LangBosnian	&h1A	
LangBosnianNeutral	&h781A	
LangBreton	&h7E	
LangBulgarian	&h02	
LangCatalan	&h03	
LangChinese	&h04	
LangChineseSimplified	&h04	
LangChineseTraditional	&h7C04	
LangCorsican	&h83	
LangCroatian	&h1A	
LangCzech	&h05	
LangDanish	&h06	
LangDari	&h8C	
LangDivehi	&h65	
LangDutch	&h13	
LangEnglish	&h09	
LangEstonian	&h25	
LangFaeroese	&h38	
LangFarsi	&h29	
LangFilipino	&h64	
LangFinnish	&h0B	
LangFrench	&h0C	
LangFrisian	&h62	
LangGalician	&h56	
LangGeorgian	&h37	
LangGerman	&h07	
LangGreek	&h08	
LangGreenlandic	&h6F	
LangGujarati	&h47	
LangHausa	&h68	
LangHebrew	&h0D	
LangHindi	&h39	
LangHungarian	&h0E	
LangIcelandic	&h0F	
LangIgbo	&h70	
LangIndonesian	&h21	
LangInuktitut	&h5D	
LangInvariant	&h7F	
LangIrish	&h3C	
LangItalian	&h10	
LangJapanese	&h11	
LangKannada	&h4B	
LangKashmiri	&h60	
LangKazak	&h3F	
LangKhmer	&h53	
LangKiche	&h86	

Sub Language Constants.

Constant	Value	Description
SublangAfrikaansSouthAfrica	&h01	
SublangAlbanianAlbania	&h01	
SublangAlsatianFrance	&h01	
SublangAmharicEthiopia	&h01	
SublangArabicAlgeria	&h05	
SublangArabicBahrain	&h0F	
SublangArabicEgypt	&h03	
SublangArabicIraq	&h02	
SublangArabicJordan	&h0B	
SublangArabicKuwait	&h0D	
SublangArabicLebanon	&h0C	
SublangArabicLibya	&h04	
SublangArabicMorocco	&h06	
SublangArabicOman	&h08	
SublangArabicQatar	&h10	
SublangArabicSaudiArabia	&h01	
SublangArabicSyria	&h0A	
SublangArabicTunisia	&h07	
SublangArabicUae	&h0E	
SublangArabicYemen	&h09	
SublangArmenianArmenia	&h01	
SublangAssameseIndia	&h01	
SublangAzeriCyrillic	&h02	
SublangAzeriLatin	&h01	
SublangBashkirRussia	&h01	
SublangBasqueBasque	&h01	
SublangBelarusianBelarus	&h01	
SublangBengaliBangladesh	&h02	
SublangBengaliIndia	&h01	
SublangBosnianBosniaHerzegovinaCyrillic	&h08	
SublangBosnianBosniaHerzegovinaLatin	&h05	
SublangBretonFrance	&h01	
SublangBulgarianBulgaria	&h01	
SublangCatalanCatalan	&h01	
SublangChineseHongkong	&h03	
SublangChineseMacau	&h05	
SublangChineseSimplified	&h02	
SublangChineseSingapore	&h04	
SublangChineseTraditional	&h01	
SublangCorsicanFrance	&h01	
SublangCroatianBosniaHerzegovinaLatin	&h04	
SublangCroatianCroatia	&h01	
SublangCustomDefault	&h03	
SublangCustomUnspecified	&h04	
SublangCzechCzechRepublic	&h01	
SublangDanishDenmark	&h01	
SublangDariAfghanistan	&h01	
SublangDefault	&h01	
SublangDivehiMaldives	&h01	
SublangDutch	&h01	
SublangDutchBelgian	&h02	
SublangEnglishAus	&h03	
SublangEnglishBelize	&h0A	
SublangEnglishCan	&h04	
SublangEnglishCaribbean	&h09	
SublangEnglishEire	&h06	
SublangEnglishIndia	&h10	
SublangEnglishJamaica	&h08	

33.23   class WindowsKeyFilterMBS

33.23.1   class WindowsKeyFilterMBS

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to filter several windows keyboard shortcuts with a low level event filter.
Notes: There is no need to keep the instance as all internal data is stored in global variables.

Key codes for Windows:

Blog Entries

- News from the MBS Xojo Plugins in version 21.5
- MBS Xojo Plugins, version 21.5pr5

33.23.2   Methods

33.23.3   Install as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Install the event filter.
Notes: Returns true on success and false on failure.
Second call in an application will return false unless Uninstall was called between.

33.23.4   Uninstall as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Uninstalls the event filter.
Notes: You only call this after Install returned true.

33.23.5   Properties

33.23.6   BlockAlt as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the alt key.
Notes: (Read and Write property)

33.23.7 BlockAltEscape as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Alt-Escape key combination.
Notes: (Read and Write property)

33.23.8 BlockAltF4 as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the ALT-F4 key combination.
Notes: By default this one closes the application.
(Read and Write property)

33.23.9 BlockAltTab as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the ALT-Tab key combination.
Notes: This combination is for task switching.
(Read and Write property)

33.23.10 BlockApplicationWindowsKey as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the application windows key.
Notes: (Read and Write property)

33.23.11 BlockBack as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the back key.
Notes: (Read and Write property)

33.23.12   BlockCancel as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Cancel key.
Notes: (Read and Write property)

33.23.13   BlockCapital as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the CapsLock key.
Notes: (Read and Write property)

33.23.14   BlockClear as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Clear key.
Notes: (Read and Write property)

33.23.15   BlockControl as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the control key.
Notes: (Read and Write property)

33.23.16   BlockControlAltDelete as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Control-Alt-Delete key combination.
Notes: Sorry, does not have any effect as this key combination is processed earlier in event chain.
(Read and Write property)

33.23.17   BlockControlEscape as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Control-Escape key combination.
Notes: (Read and Write property)

33.23.18   BlockDelete as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the delete key.
Notes: (Read and Write property)

33.23.19   BlockDown as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the down key.
Notes: (Read and Write property)

33.23.20   BlockEnd as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the end key.
Notes: (Read and Write property)

33.23.21   BlockEscape as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the escape key.
Notes: (Read and Write property)

33.23.22   BlockExecute as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the execute key.
Notes: (Read and Write property)

33.23.23   BlockF1 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F1 key.
Notes: (Read and Write property)

33.23.24   BlockF10 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F10 key.
Notes: (Read and Write property)

33.23.25   BlockF11 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F11 key.
Notes: (Read and Write property)

33.23.26   BlockF12 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F12 key.
Notes: (Read and Write property)

33.23.27   BlockF13 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F13 key.
Notes: (Read and Write property)

33.23.28   BlockF14 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F14 key.
Notes: (Read and Write property)

33.23.29   BlockF15 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F15 key.
Notes: (Read and Write property)

33.23.30   BlockF16 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F16 key.
Notes: (Read and Write property)

33.23.31   BlockF17 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F17 key.
Notes: (Read and Write property)

33.23.32   BlockF18 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F18 key.
Notes: (Read and Write property)

33.23.33   BlockF19 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F19 key.
Notes: (Read and Write property)

33.23.34   BlockF2 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F2 key.
Notes: (Read and Write property)

33.23.35   BlockF20 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F20 key.
Notes: (Read and Write property)

33.23.36   BlockF21 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F21 key.
Notes: (Read and Write property)

33.23.37   BlockF22 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F22 key.
Notes: (Read and Write property)

33.23.38   BlockF23 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F23 key.
Notes: (Read and Write property)

33.23.39   BlockF24 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F24 key.
Notes: (Read and Write property)

33.23.40   BlockF3 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F3 key.
Notes: (Read and Write property)

33.23.41   BlockF4 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F4 key.
Notes: (Read and Write property)

33.23.42   BlockF5 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F5 key.
Notes: (Read and Write property)

33.23.43   BlockF6 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F6 key.
Notes: (Read and Write property)

33.23.44   BlockF7 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F7 key.
Notes: (Read and Write property)

33.23.45 BlockF8 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F8 key.
Notes: (Read and Write property)

33.23.46 BlockF9 as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the F9 key.
Notes: (Read and Write property)

33.23.47 BlockHelp as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the help key.
Notes: (Read and Write property)

33.23.48 BlockHome as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the home key.
Notes: (Read and Write property)

33.23.49 BlockInsert as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the insert key.
Notes: (Read and Write property)

33.23.50 BlockLeft as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the left key.
Notes: (Read and Write property)

33.23.51   BlockLeftButton as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the left mouse button.
Notes: (Read and Write property)

33.23.52   BlockLeftWindowsKey as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the left windows key.
Notes: (Read and Write property)

33.23.53   BlockMiddleButton as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the middle mouse button.
Notes: (Read and Write property)

33.23.54   BlockPause as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the pause key.
Notes: (Read and Write property)

33.23.55   BlockPrint as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the print key.
Notes: BlockSnapshot is to block screenshots.
(Read and Write property)

33.23.56 BlockReturn as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the return key.
Notes: (Read and Write property)

33.23.57 BlockRight as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the right key.
Notes: (Read and Write property)

33.23.58 BlockRightButton as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the right mouse button.
Notes: (Read and Write property)

33.23.59 BlockRightWindowsKey as boolean

Plugin Version: 6.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the right windows key.
Notes: (Read and Write property)

33.23.60 BlockSelect as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the select key.
Notes: (Read and Write property)

33.23.61 BlockShift as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the shift key.
Notes: (Read and Write property)

33.23.62   BlockShiftSpace as boolean

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the Shift-Space key combination.
Notes: This combination is for making non-breakable spaces.
(Read and Write property)

33.23.63   BlockSleep as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the sleep key.
Notes: (Read and Write property)

33.23.64   BlockSnapshot as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the snapshot key.
Notes: This is the print key to do screenshots.
(Read and Write property)

33.23.65   BlockSpace as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the space key.
Notes: (Read and Write property)

33.23.66   BlockTab as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the tab key.
Notes: (Read and Write property)

33.23.67 BlockUp as Boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to filter the up key.
Notes: (Read and Write property)

33.23.68 BlockKey(virtualkeycode as Integer) as boolean

Plugin Version: 7.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to block a key with a specific virtual key code.
Notes: You can set this to true for all keys you want to block.
The virtual key codes are in the windows header files.
For most common keys this class has properties. More can be added.
(Read and Write computed property)

33.23.69 Events

33.23.70 KeyDown(vkCode as Integer, scanCode as Integer, flags as Integer, time as Integer) as Boolean

Plugin Version: 10.2, Platform: Windows, Targets: .

Function: The key down event.
Notes: vkCode: Specifies a virtual-key code. The code must be a value in the range 1 to 254.
scanCode: Specifies a hardware scan code for the key.
flags: Specifies the extended-key flag, event-injected flag, context code, and transition-state flag. This member is specified as follows.
time: Specifies the time stamp for this message.

the flag bits are like this:
bit 0: Specifies whether the key is an extended key, such as a function key or a key on the numeric keypad. The value is 1 if the key is an extended key; otherwise, it is 0.
bit 4: Specifies whether the event was injected. The value is 1 if the event was injected; otherwise, it is 0.
bit 5: Specifies the context code. The value is 1 if the ALT key is pressed; otherwise, it is 0.
bit 7: Specifies the transition state. The value is 0 if the key is pressed and 1 if it is being released.
Other bits are reserved.

There can be only one WindowsKeyFilterMBS object in your application which receives this events.

We have currently no idea in what context this event is received. And you should return as soon as possible as this event may be called a lot of times and can block user input.

Return true if you handled the key and false if you did not handle it.
Please have this event being as fast as possible. Do not put msgbox there. Better detect a key shortcut and launch a timer to perform the action a millisecond later, so you don't block the keyboard event system.
Blocks only KeyUp or only KeyDown can lead to trouble if the internal counters for special keys get our of sync.

33.23.71   KeyUp(vkCode as Integer, scanCode as Integer, flags as Integer, time as Integer) as Boolean

Plugin Version: 10.2, Platform: Windows, Targets: .

Function: The key up event.
Notes: vkCode: Specifies a virtual-key code. The code must be a value in the range 1 to 254.
scanCode: Specifies a hardware scan code for the key.
flags: Specifies the extended-key flag, event-injected flag, context code, and transition-state flag. This member is specified as follows.
time: Specifies the time stamp for this message.

the flag bits are like this:
bit 0: Specifies whether the key is an extended key, such as a function key or a key on the numeric keypad. The value is 1 if the key is an extended key; otherwise, it is 0.
bit 4: Specifies whether the event was injected. The value is 1 if the event was injected; otherwise, it is 0.
bit 5: Specifies the context code. The value is 1 if the ALT key is pressed; otherwise, it is 0.
bit 7: Specifies the transition state. The value is 0 if the key is pressed and 1 if it is being released.
Other bits are reserved.

There can be only one WindowsKeyFilterMBS object in your application which receives this events.

We have currently no idea in what context this event is received. And you should return as soon as possible as this event may be called a lot of times and can block user input.

Return true if you handled the key and false if you did not handle it.
Please have this event being as fast as possible. Do not put msgbox there. Better detect a key shortcut and launch a timer to perform the action a millisecond later, so you don't block the keyboard event system.
Blocks only KeyUp or only KeyDown can lead to trouble if the internal counters for special keys get our of sync.

VK_LBUTTON	&h01
VK_RBUTTON	&h02
VK_CANCEL	&h03
VK_MBUTTON	&h04
VK_XBUTTON1	&h05 (Windows 2000 and newer)
VK_XBUTTON2	&h06 (Windows 2000 and newer)
VK_BACK	&h08
VK_TAB	&h09
VK_CLEAR	&h0C
VK_RETURN	&h0D
VK_SHIFT	&h10
VK_CONTROL	&h11
VK_MENU	&h12
VK_PAUSE	&h13
VK_CAPITAL	&h14
VK_KANA	&h15
VK_HANGUL	&h15
VK_JUNJA	&h17
VK_FINAL	&h18
VK_HANJA	&h19
VK_KANJI	&h19
VK_ESCAPE	&h1B
VK_CONVERT	&h1C
VK_NONCONVERT	&h1D
VK_ACCEPT	&h1E
VK_MODECHANGE	&h1F
VK_SPACE	&h20
VK_PRIOR	&h21
VK_NEXT	&h22
VK_END	&h23
VK_HOME	&h24
VK_LEFT	&h25
VK_UP	&h26
VK_RIGHT	&h27
VK_DOWN	&h28
VK_SELECT	&h29
VK_PRINT	&h2A
VK_EXECUTE	&h2B
VK_SNAPSHOT	&h2C
VK_INSERT	&h2D
VK_DELETE	&h2E
VK_HELP	&h2F
VK_0 - VK_9	are the same as ASCII "0" - "9" (&h30 - &h39)
VK_A - VK_Z	are the same as ASCII "A" - "Z" (&h41 - &h5A)
VK_LWIN	&h5B
VK_RWIN	&h5C
VK_APPS	&h5D
VK_SLEEP	&h5F
VK_NUMPAD0	&h60
VK_NUMPAD1	&h61
VK_NUMPAD2	&h62
VK_NUMPAD3	&h63
VK_NUMPAD4	&h64
VK_NUMPAD5	&h65

33.24　class WindowsListMBS

33.24.1　class WindowsListMBS

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to list all the windows on a Windows system.
Notes: To find all processes on Mac, use the ProcessMBS class.
To find all windows on Mac OS X, use the CGSWindowListMBS class.
Blog Entries

- MBS Xojo Plugins, version 19.2pr1

- SetFrontMost for Windows

- MBS Xojo / Real Studio Plugins, version 16.4pr1

33.24.2　Methods

33.24.3　ActivateWindow(index as integer)

Plugin Version: 19.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Activates the window.
Notes: This function does more than just show. if the window has been minimized, it will restore it. Also the window is moved to front and made the current window for keyboard input.

Sadly windows limits which windows can go to front, so some windows may not come to front unless they are clicked on.

As activation can be asynchronously, the window may activate later, e.g. after current method ends.

33.24.4　Constructor

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor which queries the values.
See also:

33.24.5 Constructor(win as DesktopWindow)

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

Function: The constructor which queries the values.
Notes: If win is not nil, we lookup child windows of this window.
See also:

33.24.6 Constructor(win as window)

Plugin Version: 16.4, Platform: Windows, Targets: Desktop only.

Function: The constructor which queries the values.
Notes: If win is not nil, we lookup child windows of this window.
See also:

33.24.7 Constructor(WindowHandle as Integer)

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor which queries the values.
Notes: If WindowHandle is not zero, we lookup child windows of this window.
See also:

33.24.8 Focus as Integer

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the handle to the window that has the keyboard focus, if the window is attached to the calling thread's message queue.

Notes: If the calling thread's message queue does not have an associated window with the keyboard focus, the return value is 0.

33.24.9   ForegroundWindow as Integer

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves a handle to the foreground window (the window with which the user is currently working).
Notes: The system assigns a slightly higher priority to the thread that creates the foreground window than it does to other threads.
The return value is a handle to the foreground window. The foreground window can be 0 in certain circumstances, such as when a window is losing activation.

33.24.10   Update

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the values and returns true on success.

33.24.11   WindowClassName(index as Integer) as string

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the window class name of the window with the given index.
Notes: For example "SciCalc" for the calc.exe main window.

This may send a message to the thread owning the window and if that one doesn't answer quickly, the call may wait for an answer.
You may want to skip your own process windows to not wait for your own threads.

33.24.12   WindowClassNameFromHandle(Handle as Integer) as String

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the window class name of the window with the given handle.
Notes: For example "SciCalc" for the calc.exe main window.

33.24.13 WindowHandle(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.14 WindowHeight(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.15 WindowIconic(index as Integer) as boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the window with the given index is minimized.

33.24.16 WindowImageFileName(index as Integer) as string

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the path to the execuable file of the process which owns the window with the given index.
Notes: Requires Windows XP or newer.
Returns an empty string on any error.

33.24.17 WindowLeft(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The x position of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.18 WindowProcessID(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The process ID of the process which ows the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.19   WindowText(index as Integer) as string

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the title of the window with the given index.
Notes: This may send a message to the thread owning the window and if that one doesn't answer quickly, the call may wait for an answer.
You may want to skip your own process windows to not wait for your own threads.

33.24.20   WindowTextFromHandle(Handle as Integer) as String

Plugin Version: 15.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the title of the window with the given handle.
Notes: This may send a message to the thread owning the window and if that one doesn't answer quickly, the call may wait for an answer.
You may want to skip your own process windows to not wait for your own threads.

33.24.21   WindowThreadHandle(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The thread handle of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.22   WindowTop(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y position of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.23   WindowVisible(index as Integer) as boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the window with the given index is visible.

33.24.24   WindowWidth(index as Integer) as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the window with the given index.
Notes: This value is updated by the Constructor and the Update method.

33.24.25   WindowZoomed(index as Integer) as boolean

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the window with the given index is zoomed.

33.24.26   Properties

33.24.27   CurrentProcessID as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current process ID.
Notes: This value is updated by the Constructor and the Update method.
(Read and Write property)

33.24.28   CurrentThreadID as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current thread ID.
Notes: This value is updated by the Constructor and the Update method.
(Read and Write property)

33.24.29   DesktopWindowHandle as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the desktop window.
Notes: This value is updated by the Constructor and the Update method.

(Read and Write property)

33.24.30   ForegroundWindowHandle as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the foreground window.
Notes: This value is updated by the Constructor and the Update method.
(Read and Write property)

33.24.31   ParentWindowHandle as Integer

Plugin Version: 16.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the parent window.
Notes: This value is set by the Constructor.
(Read and Write property)

33.24.32   WindowCount as Integer

Plugin Version: 8.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of windows.
Notes: This value is updated by the Constructor and the Update method.
(Read and Write property)

33.25 class WindowsMonitorMBS

33.25.1 class WindowsMonitorMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to query monitor details on Windows.
Example:

for each m as WindowsMonitorMBS in WindowsMonitorMBS.AllMonitors
MsgBox m.DeviceName+": "+str(m.Left)+"x"+str(m.top)+" "+str(m.Width)+"x"+str(m.Height)
next

Blog Entries

- MBS Real Studio Plugins, version 11.3pr2

33.25.2 Methods

33.25.3 AllMonitors as WindowsMonitorMBS()

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all monitors.
Example:

for each m as WindowsMonitorMBS in WindowsMonitorMBS.AllMonitors
MsgBox m.DeviceName+": "+str(m.Left)+"x"+str(m.top)+" "+str(m.Width)+"x"+str(m.Height)
next

Notes: Returns an array with an entry for each monitor connected to this computer.
See also:

- 33.25.4 AllMonitors(monitors() as WindowsMonitorMBS) as Integer 1163

33.25.4 AllMonitors(monitors() as WindowsMonitorMBS) as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries all monitors.
Notes: Returns the number of monitors found and sets the monitor objects in the array.
if the array is too short, you get the first ubound(monitors)+1 fonts set there.
See also:

33.25.5 MonitorFromPoint(x as Integer, y as Integer, flags as Integer = 0) as WindowsMonitorMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The MonitorFromPoint function retrieves a handle to the display monitor that contains a specified point.
Example:

```
dim m as WindowsMonitorMBS = WindowsMonitorMBS.MonitorFromPoint(1900, 1000)

if m = nil then
MsgBox "Your display is smaller than 1900 x 1000"
else
MsgBox m.DeviceName
end if
```

Notes: x and y: The point of interest in virtual-screen coordinates.
Flags: Determines the function's return value if the point is not contained within any display monitor.

This parameter can be one of the following values.

Value	Meaning
kDefaultToNearest	Returns a handle to the display monitor that is nearest to the point.
kDefaultToNull	Returns nil.
kDefaultToPrimary	Returns a handle to the primary display monitor.

If the point is contained by a display monitor, the return value is that display monitor.
If the point is not contained by a display monitor, the return value depends on the value of flags.

33.25.6 MonitorFromRect(left as Integer, top as Integer, width as Integer, height as Integer, flags as Integer = 0) as WindowsMonitorMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The MonitorFromRect function retrieves the display monitor that has the largest area of intersection with a specified rectangle.
Example:

```
dim m as WindowsMonitorMBS = WindowsMonitorMBS.MonitorFromRect(0,0,100,100)
```

MsgBox m.DeviceName

Notes: left, top, width, height: The rectangle of interest in virtual-screen coordinates.
Flags: Determines the function's return value if the rectangle does not intersect any display monitor.

This parameter can be one of the following values.

Value	Meaning
kDefaultToNearest	Returns a handle to the display monitor that is nearest to the rectangle.
kDefaultToNull	Returns nil.
kDefaultToPrimary	Returns a handle to the primary display monitor.

If the rectangle intersects one or more display monitor rectangles, the return value is the display monitor that has the largest area of intersection with the rectangle.
If the rectangle does not intersect a display monitor, the return value depends on the value of Flags.

33.25.7   MonitorFromWindow(win as DesktopWindow, flags as integer = 0) as WindowsMonitorMBS

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

Function: The MonitorFromWindow function retrieves the display monitor that has the largest area of intersection with the bounding rectangle of a specified window.
Notes: win: The window of interest.
Flags: Determines the function's return value if the window does not intersect any display monitor.

This parameter can be one of the following values:

Value	Meaning
kDefaultToNearest	Returns the display monitor that is nearest to the window.
kDefaultToNull	Returns nil.
kDefaultToPrimary	Returns the primary display monitor.

If the window intersects one or more display monitor rectangles, the return value is the display monitor that has the largest area of intersection with the window.

If the window does not intersect a display monitor, the return value depends on the value of Flags.
If the window is currently minimized, MonitorFromWindow uses the rectangle of the window before it was

minimized.
See also:

33.25.8   MonitorFromWindow(win as window, flags as Integer = 0) as Windows-MonitorMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop only.

Function: The MonitorFromWindow function retrieves the display monitor that has the largest area of intersection with the bounding rectangle of a specified window.
Example:

dim m as WindowsMonitorMBS = WindowsMonitorMBS.MonitorFromWindow(window1)
MsgBox m.DeviceName

Notes: win: The window of interest.
Flags: Determines the function's return value if the window does not intersect any display monitor.

This parameter can be one of the following values:

Value	Meaning
kDefaultToNearest	Returns the display monitor that is nearest to the window.
kDefaultToNull	Returns nil.
kDefaultToPrimary	Returns the primary display monitor.

If the window intersects one or more display monitor rectangles, the return value is the display monitor that has the largest area of intersection with the window.

If the window does not intersect a display monitor, the return value depends on the value of Flags.
If the window is currently minimized, MonitorFromWindow uses the rectangle of the window before it was minimized.
See also:

33.25.9 MonitorFromWindow(WindowHandle as Integer, flags as Integer = 0) as WindowsMonitorMBS

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The MonitorFromWindow function retrieves the display monitor that has the largest area of intersection with the bounding rectangle of a specified window.
Example:

dim m as WindowsMonitorMBS = WindowsMonitorMBS.MonitorFromWindow(window1.handle)
MsgBox m.DeviceName

Notes: WindowHandle: A handle to the window of interest.
Flags: Determines the function's return value if the window does not intersect any display monitor.

This parameter can be one of the following values:

Value	Meaning
kDefaultToNearest	Returns the display monitor that is nearest to the window.
kDefaultToNull	Returns nil.
kDefaultToPrimary	Returns the primary display monitor.

If the window intersects one or more display monitor rectangles, the return value is the display monitor that has the largest area of intersection with the window.

If the window does not intersect a display monitor, the return value depends on the value of Flags.
If the window is currently minimized, MonitorFromWindow uses the rectangle of the window before it was minimized.
See also:

- 33.25.7 MonitorFromWindow(win as DesktopWindow, flags as integer = 0) as WindowsMonitorMBS 1165

- 33.25.8 MonitorFromWindow(win as window, flags as Integer = 0) as WindowsMonitorMBS 1166

33.25.10 Properties

33.25.11 Bottom as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bottom value of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the

monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read and Write property)

33.25.12   DeviceName as String

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that specifies the device name of the monitor being used.
Notes: (Read and Write property)

33.25.13   Height as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read and Write property)

33.25.14   HMonitor as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal window handle for this monitor.
Notes: (Read and Write property)

33.25.15   IsPrimary as Boolean

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this is the primary display monitor.
Notes: (Read and Write property)

33.25.16   Left as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The left position of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.

(Read and Write property)

33.25.17 Right as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The right value of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read and Write property)

33.25.18 Top as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The top position of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read and Write property)

33.25.19 Width as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the display monitor.
Notes: Specifies the display monitor rectangle, expressed in virtual-screen coordinates. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read and Write property)

33.25.20 WorkBottom as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bottom value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read and Write property)

33.25.21   WorkHeight as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read and Write property)

33.25.22   WorkLeft as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The left value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read and Write property)

33.25.23   WorkRight as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The right value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values. (Read and Write property)

33.25.24   WorkTop as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The top value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.

(Read and Write property)

33.25.25   WorkWidth as Integer

Plugin Version: 11.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width value from useable screen area.
Notes: Specifies the work area rectangle of the display monitor that can be used by applications, expressed in virtual-screen coordinates. Windows uses this rectangle to maximize an application on the monitor. The rest of the area of the monitor contains system windows such as the task bar and side bars. Note that if the monitor is not the primary display monitor, some of the rectangle's coordinates may be negative values.
(Read and Write property)

33.25.26   Constants

Constants

Constant	Value	Description
kDefaultToNearest	2	One of the flag constants.
		Return the display monitor that is nearest to the window, point or rectangle.
kDefaultToNull	0	One of the flag constants.
		Returns nil if no monitor matches is located at point, window or rectangle.
kDefaultToPrimary	1	One of the flag constants.
		Returns the primary display monitor if point, rect or window does not point to a monitor.

33.26   class WindowsPipeMBS

33.26.1   class WindowsPipeMBS

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a Windos Pipe.
Notes: A replacement for IPCSocket on Windows.
Can be used in two applications to exchange data.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.5

- MBS Xojo Plugins, version 19.5pr1

- New Windows Pipe class for Xojo

Videos

- XDC 2020 MBS Plugins Presentation

Xojo Developer Magazine

- 18.3, page 51: Happy Birthday MonkeyBread Software, What is new in the MBS Xojo Plugins by Stefanie Juchmes

33.26.2   Methods

33.26.3   Close

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Closes the pipe.
Notes: Called by destructor automatically.

33.26.4   Constructor

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

33.26.5 CreatePipe(Name as string, MessageMode as Boolean = false, Buffer-Size as Integer = &h100000, AllowAllUsers as Boolean = false) as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a named pipe.
Notes: Buffersize defines the size for input/output buffers.
Returns true on success or false on failure.
Lasterror is set.

Normally the pipe allows only same user.
But if AllowAllUsers is true, you can talk to other users or from service to user.

If name does not start with \then we prefix it with "\\.\Pipe\".

33.26.6 OpenPipe(Name as string) as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Opens an existing named pipe from client side.
Notes: Returns true on success or false on failure.
Lasterror is set.

If name does not start with \then we prefix it with "\\.\Pipe\".

33.26.7 Peek(ByteCount as Integer) as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads data without removing it from the buffer.
Notes: Reads up to the given number of bytes.
BytesRead property is set and data is returned.
LastError property is set.

33.26.8 PeekAll as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads all data without removing it from the buffer.
Notes: Reads whole message or in byte mode

BytesRead property is set and data is returned.
LastError property is set.

33.26.9   Read(ByteCount as Integer) as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads data.
Notes: Reads up to the given number of bytes.
BytesRead property is set and data is returned.
LastError property is set.

33.26.10   ReadAll as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads all data.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Dim r As String = pipe.ReadAll
Dim d As String = DefineEncoding(r, Encodings.UTF8)

List.AddRow d

Notes: Reads whole message or in byte mode what is available.
BytesRead property is set and data is returned.
LastError property is set.

33.26.11   Write(data as MemoryBlock)

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes data.
Notes: Lasterror is set.
BytesWritten is set to number of bytes sent.
See also:

- 33.26.12 Write(data as string)   1175

33.26.12 Write(data as string)

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes data.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Dim d As String = ConvertEncoding(MessageField.Text, encodings.UTF8)
If d.Len >0 Then
pipe.Write d
MessageField.Text = ""
End If

Notes: Lasterror is set.
BytesWritten is set to number of bytes sent.
See also:

- 33.26.11 Write(data as MemoryBlock) 1174

33.26.13 Properties

33.26.14 BytesAvailable as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries number of bytes available in buffer.
Notes: (Read only property)

33.26.15 BytesRead as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of bytes read.
Notes: (Read and Write property)

33.26.16 BytesWritten as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Number of bytes written.
Notes: (Read and Write property)

33.26.17   Handle as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle for the pipe.
Notes: (Read only property)

33.26.18   InputBufferSize as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The input buffer size.
Notes: (Read only property)

33.26.19   IsClient as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this is client or server side.
Notes: True if this is client side.
(Read only property)

33.26.20   IsMessageMode as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this pipe is in message mode.
Notes: (Read only property)

33.26.21   IsOpen as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries whether pipe is open.
Notes: (Read only property)

33.26.22 IsServer as Boolean

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this is client or server side.
Notes: True if this is server side.
(Read only property)

33.26.23 LastError as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: A Windows error code.
(Read and Write property)

33.26.24 MessageBytesAvailable as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries number of bytes available for current message.
Notes: (Read only property)

33.26.25 Name as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the pipe.
Notes: This is name as used in CreatePipe or OpenPipe method.
(Read and Write property)

33.26.26 NamedPipeClientComputerName as String

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the client computer name for the specified named pipe.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Msgbox "NamedPipeClientComputerName: "+pipe.NamedPipeClientComputerName

Notes: (Read only property)

33.26.27 NamedPipeClientProcessId as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the client process identifier for the specified named pipe.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Msgbox "NamedPipeClientProcessId: "+str(pipe.NamedPipeClientProcessId)

Notes: (Read only property)

33.26.28 NamedPipeClientSessionId as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the client session identifier for the specified named pipe.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Msgbox "NamedPipeClientSessionId: "+str(pipe.NamedPipeClientSessionId)

Notes: (Read only property)

33.26.29 NamedPipeServerProcessId as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the server process identifier for the specified named pipe.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Msgbox "NamedPipeServerProcessId: "+str(pipe.NamedPipeServerProcessId)

Notes: (Read only property)

33.26.30 NamedPipeServerSessionId as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves the server session identifier for the specified named pipe.
Example:

dim Pipe as WindowsPipeMBS // your pipe

Msgbox "NamedPipeServerSessionId: "+str(pipe.NamedPipeServerSessionId)

Notes: (Read only property)

33.26.31 OutputBufferSize as Integer

Plugin Version: 19.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The output buffer size.
Notes: (Read only property)

33.26.32 Events

33.26.33 Connected

Plugin Version: 19.5, Platform: Windows, Targets: .

Function: The connected event.
Notes: Called when pipe is connected on either client or server side.

33.26.34 DataAvailable(BytesAvailable as Integer, MessageBytesAvailable as Integer)

Plugin Version: 19.5, Platform: Windows, Targets: .

Function: The event called when new data is available.

33.26.35   PipeBroken

Plugin Version: 19.5, Platform: Windows, Targets: .

Function: The event called when pipe breaks.
Notes: The plugin will stop calling DataAvailable handler.
You can call Close and CreatePipe method to open pipe again.

33.27 class WindowsPreviewHandlerMBS

33.27.1 class WindowsPreviewHandlerMBS

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: The class for showing previews on windows.
Notes: This is linke a control which you can put into a window to show a preview for a file.

Uses IPreviewHandler class from Microsoft, so also see here:
https://docs.microsoft.com/en-us/windows/win32/api/shobjidl_core/nn-shobjidl_core-ipreviewhandler
https://docs.microsoft.com/de-de/windows/win32/shell/preview-handlers

The preview handlers we find are those listed in the registry here:
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\PreviewHandlers

See WinPreviewControlMBS control to use this class as a control.
Blog Entries

- MBS Xojo Plugins, version 23.3pr4

- PDF Viewer control for Windows

- MBS Xojo Plugins, version 23.3pr2

- MBS Xojo Plugins, version 22.4pr5

- MBS Xojo Plugins, version 21.6pr3

- MonkeyBread Software Releases the MBS Real Studio plug-ins in version 12.3

- MBS Real Studio Plugins, version 12.3pr11

- File Previews on Windows

Xojo Developer Magazine

- 10.6, page 10: News

33.27.2 Methods

33.27.3 Constructor(ClassID as string)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Initializes the object with loading the class with the given classid.
Notes: You find those ClassIDs in the registry (See example project).

See HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\PreviewHandlers in the registry.

33.27.4   DoPreview

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Directs the preview handler to load data from the source specified in an earlier Initialize method call, and to begin rendering to the previewer window.
Notes: Lasterror is set.

If the previewer window has not yet been created, then it must be created after this method has been called. The preview handler is responsible for painting the area specified in SetWindow or SetRect. If these methods are called while the preview handler is rendering, the window must be reparented/resized without stopping or restarting the rendering of the item.
This method should be called only after SetWindow has been called.
Aditionally, this method should be called only after Constructor has been called.

33.27.5   InitWithData(data as MemoryBlock)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.

You need to call Preview next on success to show a preview.
See also:

33.27.6   InitWithData(data as string)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.

You need to call Preview next on success to show a preview.
See also:

33.27.7   InitWithFile(file as folderitem)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given file.
Notes: Lasterror is set.
If the preview handler does not support reading file, the plugin reads the file and passes as data.

You need to call Preview next on success to show a preview.

If the file path is invalid, the preview will show a file not found error message.

33.27.8   SetBackgroundColor(red as Integer, green as Integer, blue as Integer)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Sets the background color of the preview handler.
Notes: Lasterror is set.
Not all preview handlers implement this method.

33.27.9   SetFocus

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Directs the preview handler to set focus to itself.
Notes: Lasterror is set.

33.27.10   SetFont(size as Integer, font as string)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Sets the font attributes to be used for text within the preview handler.
Notes: Lasterror is set.
Not all preview handlers implement this method.

33.27.11   SetRect(left as Integer, top as Integer, width as Integer, height as Integer)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Directs the preview handler to change the area within the parent hwnd that it draws into.
Notes: Lasterror is set.

If called before the preview handler window has been created, the new rectangle replaces the rectangle previously received in the SetWindow call.
If called after the preview handler window has been created, the preview handler window must be resized.
If the preview handler is already rendering, then the preview must be resized without interrupting the render process.

33.27.12   SetTextColor(red as Integer, green as Integer, blue as Integer)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Sets the color of the text within the preview handler.
Notes: Lasterror is set.
Not all preview handlers implement this method.

33.27.13   SetWindow(win as ContainerControl)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent of the preview to be the container with it's size.
Notes: We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.14   SetWindow(win as ContainerControl, left as integer, top as integer, width as integer, height as integer)

Plugin Version: 22.1, Platform: Windows, Targets: Desktop only.

Function: Sets the parent to a container control.
Notes: Coordinates in pixels.
See also:

33.27.15   SetWindow(win as Control)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent of the preview to be the control with it's size.
Notes: e.g. use a Canvas to host the preview.

We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.16   SetWindow(win as Control, left as integer, top as integer, width as integer, height as integer)

Plugin Version: 22.1, Platform: Windows, Targets: Desktop only.

Function: Sets the parent to a control.
Notes: Coordinates in pixels.
See also:

33.27.17 SetWindow(win as DesktopContainer)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent of the preview to be the container with it's size.
Notes: We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.18 SetWindow(win as DesktopContainer, left as integer, top as integer, width as integer, height as integer)

Plugin Version: 22.1, Platform: Windows, Targets: Desktop only.

Function: Sets the parent to a desktop container control.
Notes: Coordinates in pixels.
See also:

33.27.19   SetWindow(win as DesktopControl)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent of the preview to be the control with it's size.
Notes: e.g. use a Canvas to host the preview.

We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.20 SetWindow(win as DesktopControl, left as integer, top as integer, width as integer, height as integer)

Plugin Version: 22.1, Platform: Windows, Targets: Desktop only.

Function: Sets the parent to a desktop control.
Notes: Coordinates in pixels.
See also:

33.27.21 SetWindow(win as DesktopWindow)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent window of the previewer window.
Notes: We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.22   SetWindow(win as DesktopWindow, left as integer, top as integer, width as integer, height as integer)

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

Function: Sets the parent window of the previewer window, as well as the area within the parent to be used for the previewer window.
Notes: Pass the window and the area for the previewer.
Lasterror is set.

The preview handler is responsible for painting the entire area. If the previewer window has been created, the preview handler must associate the previewer window to the new parent hwnd and resize the previewer window to the area defined by the rectangle. If the previewer window has not yet been created, the preview handler must remember this information for when the previewer window is created in DoPreview.
It is preferred that this information be stored prior to calling DoPreview. Doing so increases performance at setup time for any cases where the preview does not start.

Coordinates in pixels.
See also:

33.27.23   SetWindow(win as Window)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent window of the previewer window.
Notes: We query size of the container automatically to set the right size independent of the screen scaling.
See also:

33.27.24   SetWindow(win as window, left as Integer, top as Integer, width as Integer, height as Integer)

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Sets the parent window of the previewer window, as well as the area within the parent to be used for the previewer window.
Notes: Pass the window and the area for the previewer.
Lasterror is set.

The preview handler is responsible for painting the entire area. If the previewer window has been created, the preview handler must associate the previewer window to the new parent hwnd and resize the previewer window to the area defined by the rectangle. If the previewer window has not yet been created, the preview handler must remember this information for when the previewer window is created in DoPreview.
It is preferred that this information be stored prior to calling DoPreview. Doing so increases performance at setup time for any cases where the preview does not start.

Coordinates in pixels.
See also:

33.27.25   Unload

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Directs the preview handler to cease rendering a preview and to release all resources that have been allocated based on the item passed in during the initialization.

Notes: Lasterror is set.
When called, the preview window will be destroyed.

This method should be called only after Constructor has been called. All resources associated with this initialization will be released. Prior to calling DoPreview, this preview handler will be re-initialized with a call to one of the initialization interfaces and a call to SetWindow.

33.27.26 Properties

33.27.27 Handle as Integer

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

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

33.27.28 Lasterror as Integer

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

33.27.29 LasterrorString as String

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: The error message for the error number in the lasterror property.
Notes: (Read only property)

33.27.30 SupportsDataLoading as Boolean

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Whether the preview handler supports loading data from a string or memoryblock.
Notes: (Read only property)

33.27.31   SupportsFileLoading as Boolean

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: Whether the preview handler supports loading files.
Notes: As the plugin can also load file and pass as data if needed, this function also returns true if SupportsDataLoading is also true.
(Read only property)

33.27.32   Window as Variant

Plugin Version: 12.3, Platform: Windows, Targets: Desktop only.

Function: The parent window.
Notes: Can reference a Window or DesktopWindow object.
(Read only property)

33.28 class WindowsPropertiesMBS

33.28.1 class WindowsPropertiesMBS

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: The class for window properties.
Example:

dim w as new WindowsPropertiesMBS(self)

dim key as string = w.EdgeGestureDisableTouchWhenFullscreen

w.Value(key) = true

dim c as integer = w.count
for i as integer = 0 to c-1
dim k as string = w.key(i)
MsgBox k+" "+w.Value(k)
next

Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.5

- MBS Xojo Plugins, version 17.5pr7

- MBS Xojo Plugins, version 17.5pr6

- Disable edge touch gestures in full screen windows

Videos

- Presentation from Munich conference about MBS Plugins.

- Presentation from London conference about MBS Plugins.

33.28.2 Methods

33.28.3 Close

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Closes properties.
Notes: Same as destructor.

33.28.4 Commit

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Commits values.
Notes: Optional as window properties are set directly.
LastError and LastErrorMessage are set.

33.28.5 Constructor(Win as DesktopWindow)

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

Function: Retrieves an object that represents a specific window's collection of properties, which allows those properties to be queried or set.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.

An application can use this function to obtain access to a window's property store so that it can set an explicit Application User Model ID (AppUserModelID) in the System.AppUserModel.ID property.

Raises FunctionNotFoundException if function is not available, PlatformNotSupportedException if not running on Windows, NilObjectException if window is nil or invalid and UnsupportedOperationException if Window properties can't be queried.
See also:

- 33.28.6 Constructor(Win as Window) 1196

33.28.6 Constructor(Win as Window)

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Retrieves an object that represents a specific window's collection of properties, which allows those properties to be queried or set.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.

An application can use this function to obtain access to a window's property store so that it can set an explicit Application User Model ID (AppUserModelID) in the System.AppUserModel.ID property.

Raises FunctionNotFoundException if function is not available, PlatformNotSupportedException if not running on Windows, NilObjectException if window is nil or invalid and UnsupportedOperationException if Window properties can't be queried.
See also:

- 33.28.5 Constructor(Win as DesktopWindow) 1196

33.28.7 Count as Integer

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Queries number of values.
Example:

```
dim w as new WindowsPropertiesMBS(self)

// show all keys with values
dim c as integer = w.count
for i as integer = 0 to c-1
dim k as string = w.key(i)
MsgBox k+" "+w.Value(k)
next
```

Notes: LastError and LastErrorMessage are set.

33.28.8 EdgeGestureDisableTouchWhenFullscreen as String

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: The property name used to disable touch gesture with full screen window.
Example:

```
dim w as new WindowsPropertiesMBS(self)
dim key as string = w.EdgeGestureDisableTouchWhenFullscreen

w.Value(key) = true
```

33.28.9 Key(Index as Integer) as String

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: The key with given index.
Notes: Index from 0 to Count-1.
Keys are passed as UUID followed by property ID.
LastError and LastErrorMessage are set.

33.28.10   Values as Dictionary

Plugin Version: 20.0, Platform: Windows, Targets: Desktop only.

Function: Queries all values as dictionary.
Notes: The dictionary is a copy, so changes to the dictionary are not going back to the window.

33.28.11   Properties

33.28.12   Handle as Integer

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

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

33.28.13   LastError as Integer

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

33.28.14   LastErrorMessage as String

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: The last error message.
Notes: (Read and Write property)

33.28.15   Value(Key as String) as Variant

Plugin Version: 17.5, Platform: Windows, Targets: Desktop only.

Function: Set or query property value.
Example:

dim w as new WindowsPropertiesMBS(self)

// show all keys with values
dim c as integer = w.count
for i as integer = 0 to c-1

dim k as string = w.key(i)
MsgBox k+" "+w.Value(k)
next

Notes: Keys are passed as UUID followed by property ID.
LastError and LastErrorMessage are set.
(Read and Write computed property)

33.29   class WindowsReportErrorMBS

33.29.1   class WindowsReportErrorMBS

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for an event source on Windows to report errors.
Example:

```
// early on app start, setup this
dim r as new WindowsReportErrorMBS("Application")

// later report something
const Category = 1
const EventID = 1000
r.Report r.kEventLogWarningType, Category, EventID, array("Hello World")
```

Blog Entries

- MBS Xojo Plugins Version 21.0 News

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.0

- MBS Xojo Plugins, version 20.6pr1

Xojo Developer Magazine

- 21.1, page 35: News from MBS Xojo Plugins, What's up with MonkeyBread Software by Stefanie Juchmes

- 19.2, page 9: News

33.29.2   Methods

33.29.3   Constructor(SourceName as String, UNCServerName as String = "")

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Retrieves a registered handle to the specified event log.
Notes: UNCServerName: The Universal Naming Convention (UNC) name of the remote server on which this operation is to be performed. If this parameter is "", the local computer is used.
SourceName: The name of the event source whose handle is to be retrieved. The source name must be a subkey of a log under the Eventlog registry key. Note that the Security log is for system use only.
Note This string must not contain characters prohibited in XML Attributes, with the exception of XML Escape sequences such as &lt &gl.

If the source name cannot be found, the event logging service uses the Application log. Although events will be reported , the events will not include descriptions because there are no message and category message files for looking up descriptions related to the event identifiers.

Raises exception on failure.

33.29.4 Report(Type as Integer, Category as Integer, EventID as Integer, Strings() as string = nil, RawData as MemoryBlock = nil)

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes an entry at the end of the event log.
Notes: Type: The type of event to be logged. See kEventLog* constants.
Category: The event category. This is source-specific information; the category can have any value.
EventID: The event identifier. The event identifier specifies the entry in the message file associated with the event source.
Strings: An array of strings that are merged into the message before Event Viewer displays the string to the user. This parameter must can be nil for no strings. Each string is limited to 31,839 characters.
RawData:

33.29.5 Properties

33.29.6 Handle as Integer

Plugin Version: 21.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal handle to the event source.
Notes: (Read and Write property)

33.29.7 Constants

Event Types

Constant	Value	Description
kEventLogAuditFailure	16	Failure Audit event
kEventLogAuditSuccess	8	Success Audit event
kEventLogErrorType	1	Error event
kEventLogInformationType	4	Information event
kEventLogSuccess	0	Information event
kEventLogWarningType	2	Warning event

33.30 class WindowsScriptErrorExceptionMBS

33.30.1 class WindowsScriptErrorExceptionMBS

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The exception class raised when an error happens while executing a script.
Example:

dim w as new WindowsScriptMBS

w.Language = "VBScript"

w.AddCode "a = 1"
w.AddCode "b = 2"

w.AllowUI = false
w.ExecuteStatement "msgbox a+b" // shows exception with access denied

Exception e as WindowsScriptErrorExceptionMBS
MsgBox e.message

Notes: Subclass of the RuntimeException class.
Blog Entries

- Windows Scripts in REALbasic

33.31 class WindowsScriptErrorMBS

33.31.1 class WindowsScriptErrorMBS

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for error details.
Example:

dim w as new WindowsScriptMBS

w.Language = "JScript"

try
w.AddCode "function Test(a,b) { 1 return a+b; } " // 1 is there to make error

dim p(-1) as string = array("Hello ", "World from JavaScript")
dim x as string = w.Run("Test", p)
catch e as WindowsScriptErrorExceptionMBS
dim error as WindowsScriptErrorMBS = w.Error
MsgBox error.Description+EndOfLine+"in "+str(error.Line)+":"+str(error.Column)
end try

33.31.2 Properties

33.31.3 Column as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Source code column position where the error occurred.
Notes: (Read and Write property)

33.31.4 Description as String

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Friendly description of error.
Notes: (Read and Write property)

33.31.5   Line as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Source code line number where the error occurred.
Notes: (Read and Write property)

33.31.6   Number as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Error number.
Notes: (Read and Write property)

33.31.7   Source as String

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Source of the error.
Notes: (Read and Write property)

33.31.8   Text as String

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Line of source code on which the error occurred.
Notes: (Read and Write property)

33.32 class WindowsScriptMBS

33.32.1 class WindowsScriptMBS

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to run Windows Scripts.
Example:

dim w as new WindowsScriptMBS

w.Language = "JScript"
w.AddCode "function Test(a,b) { return a+b; } "

dim p(-1) as string = array("Hello ", "World from JavaScript")
dim x as string = w.Run("Test", p)

MsgBox x

Notes: You may also read this MSDN page:
http://msdn.microsoft.com/en-us/library/ms756007%28VS.85%29.aspx

ScriptControl from Microsoft is 32-bit only. If you need 64-bit, please install a 64-bit alternative MSScript.ocx
file:
http://www.eonet.ne.jp/
textasciitilde gakana/tablacus/scriptcontrol_en.html
Blog Entries

- Windows Scripting

- MBS REALbasic plug-ins version 10.0

- Windows Scripts in REALbasic

33.32.2 Methods

33.32.3 AddCode(code as string)

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Add code to the global module.
Example:

dim w as new WindowsScriptMBS

w.Language = "VBScript"

w.AddCode "a = 1" // define some global variables
w.AddCode "b = 2"

Notes: If you use VBScript, you don't need "dim" keyword for defining variables.

33.32.4   ClearError

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Clears the current error.

33.32.5   Eval(code as string) as string

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Evaluate an expression within the context of the global module.

33.32.6   ExecuteStatement(statement as string)

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Execute a statement within the context of the global module.
Example:

dim w as new WindowsScriptMBS

w.Language = "VBScript"

w.AddCode "a = 1"
w.AddCode "b = 2"

w.AllowUI = true
w.ExecuteStatement "msgbox a+b" // shows 3

w.AllowUI = false
w.ExecuteStatement "msgbox a+b" // shows exception with access denied

Exception e as WindowsScriptErrorExceptionMBS
MsgBox e.message

33.32.7 Reset

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reset the scripting engine to a newly created state.

33.32.8 Run(functionName as string, parameters() as string) as string

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Call a procedure defined in the global module.
Example:

```
dim w as new WindowsScriptMBS

// First try VBScript
w.Language = "VBScript"

dim lines(-1) as string

lines.append "Function Test(a,b)"
lines.append "Test = a+b"
lines.append "End Function"

w.AddCode Join(lines, EndOfLine.Windows)

dim p(-1) as string = array("Hello ", "World from VBScript")
dim x as string = w.Run("Test", p)

'MsgBox x

// Second try JScript
w.Language = "JScript"

redim lines(-1)
lines.append "function Test(a,b)"
lines.Append " { "
lines.append "return a+b;"
lines.append " } "

w.AddCode Join(lines, EndOfLine.Windows)

p = array("Hello ", "World from JavaScript")
x = w.Run("Test", p)

MsgBox x
```

Exception e as WindowsScriptErrorExceptionMBS
MsgBox "Exception: "+e.message

33.32.9   Properties

33.32.10   AllowUI as Boolean

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enable or disable display of the UI.
Example:

dim w as new WindowsScriptMBS

w.Language = "VBScript"

w.AddCode "a = 1"
w.AddCode "b = 2"

w.AllowUI = true
w.ExecuteStatement "msgbox a+b" // shows 3

w.AllowUI = false
w.ExecuteStatement "msgbox a+b" // shows exception with access denied

Exception e as WindowsScriptErrorExceptionMBS
MsgBox e.message

Notes: (Read and Write property)

33.32.11   Error as WindowsScriptErrorMBS

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The current error status.
Notes: (Read only property)

33.32.12   Language as String

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Language engine to use.
Example:

dim w as new WindowsScriptMBS

w.Language = w.kLanguageJScript

MsgBox w.Language // shows "JScript"

w.Language = w.kLanguageVBScript

MsgBox w.Language // shows "VBScript"

Notes: Use kLanguageVBScript or kLanguageJScript constants.
Setting this property calls reset internally.
(Read and Write property)

33.32.13   Lasterror as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: 0 is no error. -1 is a parameter error on the plugin side. All other codes are from Windows.
(Read and Write property)

33.32.14   SitehWnd as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The window handle for the window being parent to any window the scripts create.
Example:

dim w as new WindowsScriptMBS
w.SitehWnd = window1.handle

Notes: The window is used as the parent window for message boxes, alerts, etc.
(Read and Write property)

33.32.15   Timeout as Integer

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Length of time in milliseconds that a script can execute before being considered hung.
Notes: (Read and Write property)

33.32.16   UseSafeSubset as Boolean

Plugin Version: 10.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Force script to execute in safe mode and disallow potentially harmful actions.
Notes: (Read and Write property)

33.32.17   Constants

Constants

Constant	Value	Description
kLanguageJScript	"JScript"	One of the language constants.
		Using this constant you can switch to JavaScript.
kLanguageVBScript	"VBScript"	One of the language constants.
		Using this constant you can switch to VBScript.

33.33 class WindowsSerialPortsMBS

33.33.1 class WindowsSerialPortsMBS

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This is a class to query the serial ports on a Windows system.
Notes: This class was created in response to feedback report 8212: Serial Port Hangup on Windows.
Blog Entries

- MBS Xojo Plugins, version 19.0pr7

- MBS Xojo / Real Studio Plugins, version 14.0pr5

- Plugins 10.1

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

33.33.2 Methods

33.33.3 Constructor(OnlyPresent as boolean = true)

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor which builds the list of devices.

33.33.4 Description(index as Integer) as string

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The description of this serial port.

33.33.5 DevicePath(index as Integer) as string

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device path of this serial port.

33.33.6   FriendlyName(index as Integer) as string

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The display name of this serial port.

33.33.7   Location(index as Integer) as string

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The location of this serial port.

33.33.8   QueryComDevices as String()

Plugin Version: 19.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries list of DOS COM Port names.
Notes: While Constructor builds a list based on registry information (setup database), this function will use QueryDosDevice to ask Windows about device names for DOS starting with COM in their name and return the list.

Returns nil in case of failure.

33.33.9   Properties

33.33.10   Count as Integer

Plugin Version: 10.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The number of serial ports found.
Notes: (Read and Write property)

33.34   class WindowsTaskbarListMBS

33.34.1   class WindowsTaskbarListMBS

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for windows with methods that control the taskbar.
Notes: It allows you to dynamically add, remove, and activate items on the taskbar.
Blog Entries

- Show progress in task bar on Windows

- SetProgressState on Windows

- MBS REALbasic plug-ins version 10.2

33.34.2   Methods

33.34.3   ActivateTab(WindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Activates an item on the taskbar.
Notes: The window is not actually activated; the window's item on the taskbar is merely displayed as active.
Lasterror is set.

33.34.4   AddTab(WindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds an item to the taskbar.
Notes: Any type of window can be added to the taskbar, but it is recommended that the window at least have the WS_CAPTION style.

Any window added with this method must be removed with the DeleteTab method when the added window is destroyed.

Lasterror is set.

33.34.5   DeleteTab(WindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes an item from the taskbar.

33.34.6   MarkFullscreenWindow(WindowHandle as Integer, Fullscreen as Boolean)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Marks a window as full-screen.
Notes: Fullscreen: A Boolean value marking the desired full-screen status of the window.

Setting the value of fFullscreen to TRUE, the Shell treats this window as a full-screen window, and the taskbar is moved to the bottom of the z-order when this window is active. Setting the value of fFullscreen to FALSE removes the full-screen marking, but does not cause the Shell to treat the window as though it were definitely not full-screen. With a FALSEfFullscreen value, the Shell depends on its automatic detection facility to specify how the window should be treated, possibly still flagging the window as full-screen.

Requires Windows XP or Windows Server 2003 or newer.
Lasterror is set.

33.34.7   RegisterTab(TabWindowHandle as Integer, MDIWindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Informs the taskbar that a new tab or document thumbnail has been provided for display in an application's taskbar group flyout.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.
By itself, registering a tab thumbnail alone will not result in its being displayed. You must also call SetTabOrder to instruct the group where to display it.

33.34.8   SetActiveAlt(WindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Marks a taskbar item as active but does not visually activate it.
Notes: SetActiveAlt marks the item associated with hwnd as the currently active item for the window's process without changing the pressed state of any item. Any user action that would activate a different tab in that process will activate the tab associated with hwnd instead. The active state of the window's item is not guaranteed to be preserved when the process associated with this window is not active. To ensure that a given tab is always active, call SetActiveAlt whenever any of your windows are activated. Calling SetActiveAlt with a nil window handle clears this state.

33.34.9 SetOverlayIcon(TabWindowHandle as Integer, IconHandle as Integer, Description as string)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Applies an overlay to a taskbar button to indicate application status or a notification to the user.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

TabWindowHandle:
The handle of the window whose associated taskbar button receives the overlay. This handle must belong to a calling process associated with the button's application and must be a valid HWND or the call is ignored.

IconHandle:
The handle of an icon to use as the overlay. This should be a small icon, measuring 16x16 pixels at 96 dpi. If an overlay icon is already applied to the taskbar button, that existing overlay is replaced.

This value can be 0. How a 0 value is handled depends on whether the taskbar button represents a single window or a group of windows.

- If the taskbar button represents a single window, the overlay icon is removed from the display.

- If the taskbar button represents a group of windows and a previous overlay is still available (received earlier than the current overlay, but not yet freed by a 0 value), then that previous overlay is displayed in place of the current overlay.

It is the responsibility of the calling application to free hIcon when it is no longer needed. This can generally be done after you've called SetOverlayIcon because the taskbar makes and uses its own copy of the icon.

Description:
A string that provides an alt text version of the information conveyed by the overlay, for accessibility purposes.

In versions of Windows earlier than Windows 7, applications often used icons in the notification area of the taskbar to display application status and notifications to the user. The Windows 7 taskbar allows an application to provide that same sort of user feedback through its taskbar button, centralizing more of the application information in one place. These overlays are similar to existing overlays used for shortcut icons or security notifications, displayed at the lower-right corner of the button.

The following illustration shows an overlay (the small, green square that indicates the user status as "Available") applied to the far-right taskbar button.

Windows Messenger taskbar button with an overlay to indicate an Available status.

Icon overlays serve as a contextual notification of status, and are intended to negate the need for a separate notification area status icon to communicate that information to the user. The application designer must decide during the development cycle which method—icon overlay or notification area status icon—best serves that application. Overlay icons are intended to supply important, long-standing status or notifications such as network status, messenger status, or new mail. They should not be frequently changed, nor should they be animated.

To display an overlay icon, the taskbar must be in the default large icon mode. If the taskbar is configured through Taskbar and Start Menu Properties to show small icons, overlays cannot be applied and calls to this method are ignored.

Because a single overlay is applied to the taskbar button instead of to the individual window thumbnails, this is a per-group feature rather than per-window. Requests for overlay icons can be received from individual windows in a taskbar group, but they do not queue. The last overlay received is the overlay shown. If the last overlay received is removed, the overlay that it replaced is restored so long as it is still active. As an example, windows 1, 2, and 3 set, in order, overlays A, B, and C. Because overlay C was received last, it is shown on the taskbar button. Window 2 calls SetOverlayIcon with a 0 value to remove overlay B. Window 3 then does the same to remove overlay C. Because window 1's overlay A is still active, that overlay is then displayed on the taskbar button.

If Windows Explorer shuts down unexpectedly, overlays are not restored when Windows Explorer is restored. The application should wait to receive the TaskbarButtonCreated message that indicates that Windows Explorer has restarted and the taskbar button has been re-created, and then call SetOverlayIcon again to reapply the overlay.

33.34.10   SetProgressState(WindowHandle as Integer, Flags as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the type and state of the progress indicator displayed on a taskbar button.
Notes: Use the ProgressStateFlag* constants. Flags that control the current state of the progress button. Specify only one of the following flags; all states are mutually exclusive of all others.

Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

Progress bar information is not shown in high contrast color schemes to guarantee that no accessibility needs are compromised.

Developers accustomed to the existing progress bar control should find the taskbar button progress indicator to be a similar experience both in concept and visuals. Here, the taskbar button itself becomes a progress bar. A taskbar button's progress indicator should be a reflection of a more detailed progress bar in the associated window. This allows the user to see specifics, such as the percentage number and the amount of

time remaining, that cannot be shown in a taskbar button. Also, because a taskbar button can show the progress of only a single window in a group, it allows the user to check the progress of individual windows. It also provides progress information to the user when the taskbar button cannot, such as in a high-contrast color scheme.

Note that a taskbar button progress bar is not intended for use with normally peripheral actions such as the loading of a Web page or the printing of a document. That type of progress should continue to be shown in a window's status bar.

The progress indicator is displayed between the taskbar button's icon or text and the background. If progress is shown for both the active taskbar button and an inactive button, shading in the respective progress bars is such that the active button is still obvious to the user. Also, button functionality such as the display of thumbnails continues to work normally when the button is being used to display progress.

When exiting an error or paused state, call this method again with the ProgressStateFlagNormal or ProgressStateFlagIndeterminate flag to continue in the original state or TBPF_NOPROGRESS if the operation is cancelled.

How the Taskbar Button Chooses the Progress Indicator for a Group

The taskbar button can show a progress indicator for only one window at a time. This includes the situation where the taskbar button represents a group and more than one window in that group is broadcasting progress information. In that case, the taskbar button chooses its progress display based on state priority. State priority is shown in the following table with priority 1 being the highest.

Priority	State
1	ProgressStateFlagError
2	ProgressStateFlagPaused
3	ProgressStateFlagNormal
4	ProgressStateFlagIndeterminate

Changing a window's state changes its priority in relation to other windows in the group which in turn might change which window in a group is used for the progress indicator in the taskbar button.

In the case of a priority collision between two windows that are broadcasting determinate progress, the window with the least progress is used.

Based on this priority, the indeterminate progress indicator can be displayed in the taskbar button only in these cases:

- The taskbar button does not represent a group and the single window that it represents has set ProgressStateFlagIndeterminate.

- The taskbar button represents a group, only one window in that group is broadcasting progress information, and that window has set ProgressStateFlagIndeterminate.

- The taskbar button represents a group, multiple windows in that group are broadcasting progress information, and all of those windows have set ProgressStateFlagIndeterminate.

A determinate progress indicator can be displayed in these cases:

- The taskbar button does not represent a group and the single window that it represents is broadcasting determinate progress information.

- The taskbar button represents a group, only one window in that group is broadcasting progress information, and that window is broadcasting determinate progress information.

- The taskbar button represents a group, multiple windows in that group are broadcasting progress information, at least one of those windows is broadcasting determinate progress information, and no window has set ProgressStateFlagError or ProgressStateFlagPaused.

Note that a call to SetProgressValue will switch a progress indicator currently in an indeterminate mode (ProgressStateFlagIndeterminate) to a normal (determinate) display and clear the ProgressStateFlagIndeterminate flag.

If the handle is a ptr, you need to cast it with Integer(YourPtr) to pass it.

33.34.11   SetProgressValue(WindowHandle as Integer, Completed as UInt64, Total as UInt64)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Displays or updates a progress bar hosted in a taskbar button to show the specific percentage completed of the full operation.
Notes: Completed: An application-defined value that indicates the proportion of the operation that has been completed at the time the method is called.
Total: An application-defined value that specifies the value ullCompleted will have when the operation is complete.

Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

Determinate Progress Bar Lifecycle

An application first calls SetProgressValue to begin the display of a determinate progress bar, and then calls it again as needed to update the bar as the progress changes. When progress is complete, the application must call SetProgressState with the TBPF_NOPROGRESS flag to dismiss the progress bar.

How the Taskbar Button Chooses the Progress Indicator for a Group

The taskbar button can show a progress indicator for only one window at a time. When the taskbar button represents a group and more than one of the windows in that group are broadcasting progress information, the taskbar button chooses its progress display based on the following state priority.

Priority	State
1	ProgressStateFlagError
2	ProgressStateFlagPaused
3	ProgressStateFlagNormal
4	ProgressStateFlagIndeterminate

Unless SetProgressState has set a blocking state (ProgressStateFlagError or ProgressStateFlagPaused) for the window, a call to SetProgressValue assumes the ProgressStateFlagNormal state even if it is not explicitly set. A call to SetProgressValue overrides and clears the ProgressStateFlagIndeterminate state.

In the case of a priority collision where two windows are broadcasting determinate progress, the window with the least progress is used.

Based on that priority, this determinate (specific percentage) progress indicator can be displayed in these cases:

- The taskbar button does not represent a group and the single window that it represents is broadcasting determinate progress information through this method.

- The taskbar button represents a group, only one window in that group is broadcasting progress information, and that window is broadcasting determinate progress information through this method.

- The taskbar button represents a group, multiple windows in that group are broadcasting progress information, at least one of those windows is broadcasting progress information through this method, and none of those windows has set the ProgressStateFlagError or ProgressStateFlagPaused state.

If a window in the group has set ProgressStateFlagError or ProgressStateFlagPaused, that state will be used for the button display. However, you can still make calls to SetProgressValue on other, unblocked windows in the group to update their progress in the background.

If the handle is a ptr, you need to cast it with Integer(YourPtr) to pass it.

33.34.12   SetTabActive(TabWindowHandle as Integer, MDIWindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Informs the taskbar that a tab or document window has been made the active window.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

TabWindowHandle: Handle of the active tab window. This handle must already be registered through RegisterTab. This value can be 0 if no tab is active.

MDIWindowHandle: Handle of the application's main window. This value tells the taskbar which group the thumbnail is a member of. This value is required and cannot be 0.

33.34.13   SetTabOrder(TabWindowHandle as Integer, InsertBeforeWindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Inserts a new thumbnail into a tabbed-document interface (TDI) or multiple-document interface (MDI) application's group flyout or moves an existing thumbnail to a new position in the application's group.
Notes: TabWindowHandle: The handle of the tab window whose thumbnail is being placed. This value is required, must already be registered through RegisterTab, and cannot be 0.
InsertBeforeWindowHandle: The handle of the tab window whose thumbnail that hwndTab is inserted to the left of. This handle must already be registered through RegisterTab. If this value is 0, the new thumbnail is added to the end of the list.

33.34.14   SetTabProperties(TabWindowHandle as Integer, flags as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Allows a tab to specify whether the main application frame window or the tab window should be used as a thumbnail or in the peek feature under certain circumstances.
Notes: An application might want to use the thumbnail or peek representation of its associated parent window if the application cannot generate its own thumbnail for a tab or for its active tab content (such as an animation) to appear live.

flags: One of the TabPropertyFlag* constants.
Available on Windows 7 or Windows Server 2008 R2 or newer.

33.34.15 SetThumbnailClip(TabWindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Selects a portion of a window's client area to display as that window's thumbnail in the taskbar.
Notes: This variant of the method clears the clip that is already in place and return to the default display of the thumbnail.

Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.
See also:

- 33.34.16 SetThumbnailClip(TabWindowHandle as Integer, x as Integer, y as Integer, w as Integer, h as Integer) 1221

33.34.16 SetThumbnailClip(TabWindowHandle as Integer, x as Integer, y as Integer, w as Integer, h as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Selects a portion of a window's client area to display as that window's thumbnail in the taskbar.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.
See also:

- 33.34.15 SetThumbnailClip(TabWindowHandle as Integer) 1221

33.34.17 SetThumbnailTooltip(TabWindowHandle as Integer, tip as string)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies or updates the text of the tooltip that is displayed when the mouse pointer rests on an individual preview thumbnail in a taskbar button flyout.
Notes: Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

TabWindowHandle: The handle to the window whose thumbnail displays the tooltip. This handle must belong to the calling process.

tip: The text to be displayed in the tooltip. This value can be empty, in which case the title of the window specified by WindowHandle is used as the tooltip.

33.34.18   UnregisterTab(TabWindowHandle as Integer)

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes a thumbnail from an application's preview group when that tab or document is closed in the application.
Notes: It is the responsibility of the calling application to free hwndTab through DestroyWindow. UnregisterTab must be called before the handle is freed.
Requires Windows 7 or Windows Server 2008 R2 or newer.
Lasterror is set.

33.34.19   Properties

33.34.20   Handle1 as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference for the ITaskbarList interface.
Notes: The ITaskbarList interface is available on Windows 2000 Professional, Windows XP or Windows 2000 Server or newer.
(Read and Write property)

33.34.21   Handle2 as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference for the ITaskbarList2 interface.
Notes: The ITaskbarList4 interface is available on Windows XP or Windows Server 2003 or newer.
(Read and Write property)

33.34.22   Handle3 as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference for the ITaskbarList3 interface.
Notes: The ITaskbarList3 interface is available on Windows 7 or Windows Server 2008 R2 or newer.
none
(Read and Write property)

33.34.23   Handle4 as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference for the ITaskbarList4 interface.
Notes: The ITaskbarList4 interface is available on Windows 7 or Windows Server 2008 R2 or newer.
(Read and Write property)

33.34.24   Lasterror as Integer

Plugin Version: 10.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code from one of the methods.
Notes: (Read and Write property)

33.34.25   Constants

Constants

Constant	Value	Description
ProgressStateFlagError	4	One of the progress state flag constants. The progress indicator turns red to show that an error has occ the windows that is broadcasting progress. This is a determina progress indicator is in the indeterminate state, it switches to a display of a generic percentage not indicative of actual progres
ProgressStateFlagIndeterminate	1	One of the progress state flag constants. The progress indicator does not grow in size, but cycles repea length of the taskbar button. This indicates activity without proportion of the progress is complete. Progress is taking pla no prediction as to how long the operation will take.
ProgressStateFlagNoProgress	0	One of the progress state flag constants. Stops displaying progress and returns the button to its norm this method with this flag to dismiss the progress bar when t complete or cancelled.
ProgressStateFlagNormal	2	One of the progress state flag constants. The progress indicator grows in size from left to right in pr estimated amount of the operation completed. This is a deter indicator; a prediction is being made as to the duration of the
ProgressStateFlagPaused	8	One of the progress state flag constants. he progress indicator turns yellow to show that progress is curre one of the windows but can be resumed by the user. No error and nothing is preventing the progress from continuing. This i state. If the progress indicator is in the indeterminate state a yellow determinate display of a generic percentage not indi progress.
TabPropertyFlagNone	0	One of the flag values you can use with SetTabProperties. No specific property values are specified. The default behavior window provides a thumbnail and peek image, either live or sta ate.
TabPropertyFlagUseAppPeekAlways	4	One of the flag values you can use with SetTabProperties. Always use the peek image provided by the main application rather than a peek image provided by the individual tab w combine this value with TabPropertyFlagUseAppPeekWhenA will result in an error.
TabPropertyFlagUseAppPeekWhenActive	8	One of the flag values you can use with SetTabProperties. When the application tab is active and a live representation available, show the main application frame in the peek feature. use the tab window. Do not combine this value with TabPr AppPeekAlways; doing so will result in an error.
TabPropertyFlagUseAppThumbnailAlways	1	One of the flag values you can use with SetTabProperties. Always use the thumbnail provided by the main application rather than a thumbnail provided by the individual tab windo bine this value with TabPropertyFlagUseAppThumbnailWhen will result in an error.
TabPropertyFlagUseAppThumbnailWhenActive	2	One of the flag values you can use with SetTabProperties. When the application tab is active and a live representation available, use the main application frame window thumbnail. use the tab window thumbnail. Do not combine this value w tyFlagUseAppThumbnailAlways; doing so will result in an err

33.35 class WindowsThreadExecutionStateMBS

33.35.1 class WindowsThreadExecutionStateMBS

Plugin Version: 23.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the thread execution state.
Example:

dim flags as integer = WindowsThreadExecutionStateMBS.kSystemRequired
dim w as new WindowsThreadExecutionStateMBS(flags)
window1.WindowsThreadExecutionState = w // store it in property

Notes: The state is set in constructor and reset in destructor, so you can put this object in a property of a window, so when the window is destroyed, the state is reset automatically.

Enables an application to inform the system that it is in use, thereby preventing the system from entering sleep or turning off the display while the application is running.

For macOS, please check NSProcessInfoMBS class.

The system automatically detects activities such as local keyboard or mouse input, server activity, and changing window focus. Activities that are not automatically detected include disk or CPU activity and video display.

To run properly on a power-managed computer, applications such as fax servers, answering machines, backup agents, and network management applications must use both kSystemRequired when they process events. Multimedia applications, such as video players and presentation applications, must use kDisplayRequired when they display video for long periods of time without user input. Applications such as word processors, spreadsheets, browsers, and games do not need to use this.

The kAwayModeRequired value should be used only when absolutely necessary by media applications that require the system to perform background tasks such as recording television content or streaming media to other devices while the system appears to be sleeping. Applications that do not require critical background processing or that run on portable computers should not enable away mode because it prevents the system from conserving power by entering true sleep.

To enable away mode, an application uses both kAwayModeRequired; to disable away mode, let the destructor reset. When away mode is enabled, any operation that would put the computer to sleep puts it in away mode instead. The computer appears to be sleeping while the system continues to perform tasks that do not require user input. Away mode does not affect the sleep idle timer; to prevent the system from entering sleep when the timer expires, an application must also set the kSystemRequired value.

This class cannot be used to prevent the user from putting the computer to sleep. Applications should respect that the user expects a certain behavior when they close the lid on their laptop or press the power button.

This function does not stop the screen saver from executing.
Blog Entries

- News from the MBS Xojo Plugins in Version 23.0

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.0

- MBS Xojo Plugins, version 22.6pr4

Xojo Developer Magazine

- 21.3, page 10: News

- 21.2, page 9: News

33.35.2   Methods

33.35.3   Constructor(Flags as Integer)

Plugin Version: 23.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the state and stores old state in property.
Example:

dim flags as integer = WindowsThreadExecutionStateMBS.kSystemRequired
dim w as new WindowsThreadExecutionStateMBS(flags)
window1.WindowsThreadExecutionState = w // store it in property

Notes: Does nothing if called on macOS, iOS or Linux.

33.35.4   Destructor

Plugin Version: 23.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.
Notes: Puts back the old state.
Does nothing if called on macOS, iOS or Linux.

33.35.5 Properties

33.35.6 newState as Integer

Plugin Version: 23.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The new state set.
Notes: (Read only property)

33.35.7 oldState as Integer

Plugin Version: 23.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The old state before.
Notes: (Read only property)

33.35.8 Constants

Flags

Constant	Value	Description
kAwayModeRequired	64	Enables away mode. Away mode should be used only by media-recording and media-distribution applications that must perform critical background processing on desktop computers while the computer appears to be sleeping.
kDisplayRequired	2	Forces the display to be on by resetting the display idle timer.
kSystemRequired	1	Forces the system to be in the working state by resetting the system idle timer.

33.36　class WindowsWMIMBS

33.36.1　class WindowsWMIMBS

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to run queries in the Windows Management Interface.
Example:

dim w as WindowsWMIMBS

w=new WindowsWMIMBS

if w.ConnectServer("root\cimv2") then
MsgBox "ConnectServer: ok"

if w.query("WQL","SELECT * FROM Win32_Processor") then
MsgBox "query: ok"

if w.NextItem then
MsgBox "NextItem: ok"

MsgBox w.GetPropertyString("Name") // string
MsgBox str(w.GetPropertyInteger("MaxClockSpeed")) // uint32
MsgBox str(w.GetPropertyInteger("ProcessorType")) // uint16

else
MsgBox "NextItem: fail"
end if
else
MsgBox "query: fail"
end if

else
MsgBox "ConnectServer: fail"
end if

Notes: In Windows 8.1 (or later) it looks like WindowsWMIMBS.InitSecurity(false) must be called at app.Constructor time as Xojo will do some things in background when opening first window which block our queries.
Subclass of the WMIObjectMBS class.
Blog Entries

- Xojo News

- Windows Task Monitor written in Xojo

- MBS Real Studio Plugins, version 12.2pr6

- Detecting drive letter for an USB Stick

- MBS Releases the MBS Real Studio plug-ins in version 12.0

- MBS Real Studio Plugins, version 11.4pr1

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr13

33.36.2   Methods

33.36.3   CancelAsyncCall as boolean

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Stops events being delivered to this WindowsWMIMBS object.
Notes: Lasterror is set.

33.36.4   ConnectServer(NetworkResource as string, Username as string="", Password as string="", Locale as string="", Authority as string="") as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Connect to WMI server.
Example:

dim w as WindowsWMIMBS // your WMI object
call w.ConnectServer("root\cimv2")

Notes: Returns true on success and false on failure.
Authority parameter added in plugin version 12.2.

33.36.5   Constructor

Plugin Version: 12.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

33.36.6   ExecNotificationQueryAsync(QueryLanguage as string, QueryText as string) as boolean

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ExecNotificationQueryAsync method executes a query to receive events.
Notes: The plugin receives events and enqueues them in a queue. You read objects from this queue in a timer and process them.
NextItem returns nil if the queue is empty.

Returns true on success and false on failure.
Lasterror is set. e.g. &h80041018 if you have syntax error in the query or &h80041058 if the query can't be parsed.

33.36.7   InitAuthentication(User as string, Domain as string, Password as string) as boolean

Plugin Version: 12.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initializes authentication data.
Notes: Call before ConnectServer.
This settings are passed to objects for authentication.

33.36.8   InitSecurity(AuthnLevel as Integer, ImpLevel as Integer) as boolean

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Registers security and sets the default security values for the process.
Notes: You can call this method before creating WindowsWMIMBS object, you can prepare security settings for remote connections.

AuthnLevel:
The default authentication level for the process. Both servers and clients use this parameter when they call CoInitializeSecurity. COM will fail calls that arrive with a lower authentication level. By default, all proxies will use at least this authentication level. This value should contain one of the authentication level constants. By default, all calls to IUnknown are made at this level.

ImpLevel:
The default impersonation level for proxies. The value of this parameter is used only when the process is a client. It should be a value from the impersonation level constants, except for kImpersonationLevelDefault, which is not for use with CoInitializeSecurity.
Outgoing calls from the client always use the impersonation level as specified. (It is not negotiated.) Incoming calls to the client can be at any impersonation level. By default, all IUnknown calls are made with this

impersonation level, so even security-aware applications should set this level carefully. To determine which impersonation levels each authentication service supports, see the description of the authentication services in COM and Security Packages. For more information about impersonation levels, see Impersonation.

Please call in App.Constructor before Xojo can initialize COM and prevent our function from working!
See also:

- 33.36.9 InitSecurity(remote as boolean) as boolean   1231

33.36.9   InitSecurity(remote as boolean) as boolean

Plugin Version: 12.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Initializes security.
Notes: You can call this method before creating WindowsWMIMBS object, you can prepare security settings for remote connections.

Calling InitSecurity with remote=true will initialize security with kImpersonationLevelImpersonate. If remote=false, we use kImpersonationLevelIdentity which is the default.

Please call in App.Constructor before Xojo can initialize COM and prevent our function from working!
See also:

- 33.36.8 InitSecurity(AuthnLevel as Integer, ImpLevel as Integer) as boolean   1230

33.36.10   NextItem as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Moves to next item in the list.
Notes: You need to call this at least once to move to the first item.
Returns false if no more items are there.

33.36.11   Query(QueryLanguage as string, QueryText as string) as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Runs a query.
Example:

dim w as WindowsWMIMBS
call w.query("WQL","SELECT * FROM Win32_Processor")

Notes: Returns true on success and false on failure.

33.36.12   Properties

33.36.13   EnumeratorHandle as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the used IEnumWbemClassObject object.
Notes: (Read only property)

33.36.14   LocatorHandle as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the used IWbemLocator object.
Notes: (Read only property)

33.36.15   ServiceHandle as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle to the used IWbemServices object.
Notes: (Read only property)

33.36.16   Constants

Authentication Level

Constant	Value	Description
kAuthenticationLevelCall	3	Authenticates only at the beginning of each remote procedure call when server receives the request. Datagram transports use kAuthenticationLevelPaket instead.
kAuthenticationLevelConnect	2	Authenticates the credentials of the client only when the client establishes relationship with the server. Datagram transports always use kAuthenticationLevelPacket instead.
kAuthenticationLevelDefault	0	Tells DCOM to choose the authentication level using its normal security blanket negotiation algorithm. For more information, see Security Blanket Negotiation.
kAuthenticationLevelNone	1	Performs no authentication.
kAuthenticationLevelPacket	4	Authenticates that all data received is from the expected client.
kAuthenticationLevelPacketIntegrity	5	Authenticates and verifies that none of the data transferred between client server has been modified.
kAuthenticationLevelPacketPrivacy	6	Authenticates all previous levels and encrypts the argument value of each mote procedure call.

Impersonation Level Constants

Constant	Value	Description
kImpersonationLevelAnonymous	1	The client is anonymous to the server. The server process can impersonate the client, but the impersonation token will not contain any information and cannot be used.
kImpersonationLevelDefault	0	DCOM can choose the impersonation level using its normal security blanket negotiation algorithm. For more information, see Security Blanket Negotiation.
kImpersonationLevelDelegate	4	The server process can impersonate the client's security context while acting on behalf of the client. The server process can also make outgoing calls to other servers while acting on behalf of the client, using cloaking. The server may use the client's security context on other machines to access local and remote resources as the client. When impersonating at this level, the impersonation token can be passed across any number of computer boundaries.
kImpersonationLevelIdentity	2	The server can obtain the client's identity. The server can impersonate the client for ACL checking, but it cannot access system objects as the client.
kImpersonationLevelImpersonate	3	The server process can impersonate the client's security context while acting on behalf of the client. This level of impersonation can be used to access local resources such as files. When impersonating at this level, the impersonation token can only be passed across one machine boundary. The Schannel authentication service only supports this level of impersonation.

33.37   class WinExceptionMBS

33.37.1   class WinExceptionMBS

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to catch windows system exceptions.
Notes: This class was made to intercept crashes in a Xojo application.
Whenever the application crashes, this class will fire its event so you have a chance to report that crash to your user and maybe save a crash report.

Do not try to save data as this could overwrite good data with bad data. Remember your application already crashed. Maybe because of data corruption!

Only one instance can exist in your application.

33.37.2   Methods

33.37.3   Close

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes the exception handler from the system.

33.37.4   Properties

33.37.5   ExceptionAddress as Integer

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The address in memory where the exception was raised.
Notes: (Read only property)

33.37.6   ExceptionCode as Integer

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The code of this exception.
Notes: Normally one of the kException* constants.
(Read only property)

33.37.7 ExceptionFlags as Integer

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The exception flags.
Notes: (Read only property)

33.37.8 ExceptionIsNonContinuable as Boolean

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether this exception allows to continue in code.
Notes: For example a division by zero can be continued.
(Read only property)

33.37.9 ExceptionName as String

Plugin Version: 8.7, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name of the exception.
Notes: If ExceptionCode=kExceptionAccessViolation then returns "ExceptionAccessViolation".

This is a convenience function which you can use to show the exception to the user.
(Read only property)

33.37.10 Events

33.37.11 GotException() as Integer

Plugin Version: 8.7, Platform: Windows, Targets: .

Function: The event called when an exception was received.
Notes: Your application has crashed on this point, but instead of seeing the Microsoft dialog to tell you that the application crashed, your application has the chance to report that problem.

Return one of the following constants: kExecuteHandler, kContinueCode, kNextHandler, kExecuteHandlerNoDialog.

The plugin already removed the crash callback from the system so if the code in this event crashes again, the system dialog shows.

33.37.12 Constants

Constants

Constant	Value	Description
kContinueCode	2	One of the constants you can use for the GotException event ret This one is not recommented as it can raise the same exception a your application.
kExceptionAccessViolation	&hC0000005	One of the constants for exception codes. The thread attempts to read from or write to a virtual addres does not have access.
kExceptionArrayBoundsExceeded	&hC000008C	One of the constants for exception codes. The thread attempts to access an array element that is out of bo underlying hardware supports bounds checking.
kExceptionBreakPoint	&h80000003	One of the constants for exception codes. A breakpoint is encountered.
kExceptionDataTypeMisalignment	&h80000002	One of the constants for exception codes. The thread attempts to read or write data that is misaligned on l does not provide alignment. For example, 16-bit values must 2-byte boundaries, 32-bit values on 4-byte boundaries, and so on
kExceptionFloatDenormalOperand	&hC000008D	One of the constants for exception codes. One of the operands in a floating point operation is denormal. value is one that is too small to represent as a standard floating
kExceptionFloatDivideByZero	&hC000008E	One of the constants for exception codes. The thread attempts to divide a floating point value by a floating of 0 (zero).
kExceptionFloatInexactResult	&hC000008F	One of the constants for exception codes. The result of a floating point operation cannot be represented decimal fraction.
kExceptionFloatInvalidOperation	&hC0000090	One of the constants for exception codes. A floating point exception that is not included in this list.
kExceptionFloatOverflow	&hC0000091	One of the constants for exception codes. The exponent of a floating point operation is greater than the lowed by the corresponding type.
kExceptionFloatStackCheck	&hC0000092	One of the constants for exception codes. The stack has overflowed or underflowed, because of a floating po
kExceptionFloatUnderflow	&hC0000093	One of the constants for exception codes. The exponent of a floating point operation is less than the magn by the corresponding type.
kExceptionIllegalInstruction	&hC000001D	One of the constants for exception codes. The thread tries to execute an invalid instruction.
kExceptionInPageError	&hC0000006	One of the constants for exception codes. The thread tries to access a page that is not present, and the syste load the page. For example, this exception might occur if a netwo is lost while running a program over a network.
kExceptionIntegerDivideByZero	&hC0000094	One of the constants for exception codes. The thread attempts to divide an integer value by an integer divis
kExceptionIntegerOverflow	&hC0000095	One of the constants for exception codes. The result of an integer operation causes a carry out of the most of the result.
kExceptionInvalidDisposition	&hC0000026	One of the constants for exception codes. An exception handler returns an invalid disposition to the excepti Programmers using a high-level language such as C should never e exception.
kExceptionNonContinueableException	&hC0000025	One of the constants for exception codes. The thread attempts to continue execution after a non-continua occurs.
kExceptionPrivilegedInstruction	&hC0000096	One of the constants for exception codes. The thread attempts to execute an instruction with an operatio allowed in the current computer mode.
kExceptionSingleStep	&h80000004	One of the constants for exception codes. A trace trap or other single instruction mechanism signals that o is executed.

33.38   class WinGestureConfigMBS

33.38.1   class WinGestureConfigMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: ets and sets the configuration for enabling gesture messages and the type of this configuration.
Example:

```
// make a config for accepting pan/rotate gestures
dim c1 as new WinGestureConfigMBS

c1.ID = c1.kGestureIDPan
c1.Want = c1.kPan
c1.Block = 0

dim c2 as new WinGestureConfigMBS
c2.ID = c2.kGestureIDRotate
c2.Want = c2.kRotate
c2.Block = 0


dim configs() as WinGestureConfigMBS
configs.Append c1
configs.Append c2

// and add to window
if WinPointerEventsMBS.SetGestureConfig(window1, configs) then
MsgBox "OK"
end if
```

Notes: It is impossible to disable two-finger panning and keep single finger panning. You must set the want bits for GC_PAN before you can set them for GC_PAN_WITH_SINGLE_FINGER_HORIZONTALLY or GC_PAN_WITH_SINGLE_FINGER_VERTICALLY.
An inertia vector is included in the GID_PAN message with the GF_END flag if inertia was disabled by a call to SetGestureConfig.
When you pass this structure, the dwID member contains information for a set of gestures. This determines what the other flags will mean. If you set flags for pan messages, they will be different from those flags that are set for rotation messages.
The following table indicates the various identifiers for gestures that are supported by the dwID member of the GESTURECONFIG structure. Note that setting dwID to 0 indicates that global gesture configuration flags are set.

The following flags are used when dwID is set to 0.

Name	Value	Description
GID_ZOOM	3	Indicates configuration settings for the zoom gesture.
GID_PAN	4	Indicates the pan gesture.
GID_ROTATE	5	Indicates the rotation gesture.
GID_TWOFINGERTAP	6	Indicates the two-finger tap gesture.
GID_PRESSANDTAP	7	Indicates the press and tap gesture.

Name	Value	Description
GC_ALLGESTURES	0x00000001	Indicates all of the gestures.

The following flags are used when dwID is set to GID_ZOOM.

Name	Value	Description
GC_ZOOM	0x00000001	Indicates the zoom gesture.

The following flags are used when dwID is set to GID_PAN.

Note Setting the GID_PAN flags in SetGestureConfig will affect the default gesture handler for panning. You should not have both dwWant and dwBlock set for the same flags; this will result in unexpected behavior. See Windows Touch Gestures for more information on panning and legacy panning support; see SetGestureConfig for examples of enabling and blocking gestures.

The following flags are used when dwID is set to GID_ROTATE.

The following flags are used when dwID is set to GID_TWOFINGERTAP.

The following flags are used when dwID is set to GID_PRESSANDTAP.

Blog Entries

- Windows Touch Events

Videos

- Presentation from Munich conference about MBS Plugins.

Name	Value	Description
GC_PAN	0x00000001	Indicates all pan gestures.
GC_PAN_WITH_SINGLE_FINGER_VERTICALLY	0x00000002	Indicates vertical pans with one finger.
GC_PAN_WITH_SINGLE_FINGER_HORIZONTALLY	0x00000004	Indicates horizontal pans with one finger.
GC_PAN_WITH_GUTTER	0x00000008	Limits perpendicular movement to primary direction until a threshold is reached to break out of the gutter.
GC_PAN_WITH_INERTIA	0x00000010	Indicates panning with inertia to smoothly slow when pan gestures stop.

Name	Value	Description
GC_ROTATE	0x00000001	Indicates the rotation gesture.

33.38.2   Properties

33.38.3   Block as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The messages to disable.
Example:

```
// make a config for accepting some pan gestures
dim c as new WinGestureConfigMBS

c.ID = kGestureIDPan
c.Want = c.kPanWithSingleFingerVertically + c.kPanWithGutter + c.kPanWithInteria
c.Block = c.kPanWithSingleFingerHorizontally
```

Notes: (Read and Write property)

33.38.4   ID as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identifier for the type of configuration that will have messages enabled or disabled.
Example:

```
// make a config for accepting some pan/rotate gestures
dim c1 as new WinGestureConfigMBS

c1.ID = 0
c1.Want = c1.kAllGestures
c1.Block = 0

dim configs() as WinGestureConfigMBS
configs.Append c1
```

Name	Value	Description
GC_TWOFINGERTAP	0x00000001	Indicates the two-finger tap gesture.

Name	Value	Description
GC_PRESSANDTAP	0x00000001	Indicates the press and tap gesture.

```
// and add to window
if WinPointerEventsMBS.SetGestureConfig(window1, configs) then
MsgBox "OK"
end if
```

Notes: (Read and Write property)

33.38.5 Want as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The messages to enable.
Example:

```
// make a config for accepting all gestures
dim c as new WinGestureConfigMBS

c.ID = 0 // catch all IDs
c.Want = c.kAllGestures
```

Notes: (Read and Write property)

33.38.6 Constants

Gesture Options

Constant	Value	Description
kAllGestures	1	Indicates all of the gestures. (for ID = 0)
kPan	1	Indicates all pan gestures.
kPanWithGutter	8	Limits perpendicular movement to primary direction until a threshold reached to break out of the gutter.
kPanWithInteria	16	Indicates panning with inertia to smoothly slow when pan gestures stop.
kPanWithSingleFingerHorizontally	4	Indicates horizontal pans with one finger.
kPanWithSingleFingerVertically	2	Indicates vertical pans with one finger.
kPressAndTap	1	Indicates the press and tap gesture.
kRotate	1	Indicates the rotation gesture.
kTwoFingerTap	1	Indicates the two-finger tap gesture.
kZoom	1	Indicates the zoom gesture. (for ID = kGestureIDZoom)

Gesture IDs

Constant	Value	Description
kGestureIDBegin	1	Begin of gesture.
kGestureIDEnd	2	End of a gesture
kGestureIDPan	4	Pan gesture
kGestureIDPressAndTap	7	Press and Tap gesture
kGestureIDRotate	5	Rotate gesture
kGestureIDTwoFingerTap	6	Two-Finger Tap
kGestureIDZoom	3	Zoom gesture

33.39 class WinGestureInfoMBS

33.39.1 class WinGestureInfoMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Stores information about a gesture.
Notes: Parameters for touches:
If kFlagBegin flag is set, save Location so you have first point. And save argument to check later.

For Zoom, get second point from this class. Than you can calculate center point and zoom factor is OldArgumentsLower/NewArgumentsLower.

For Pan, check difference between last point and new point to know how much moved.

For Rotate, please check difference between old RotateAngle and new RotateAngle.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Windows Touch Events

33.39.2 Methods

33.39.3 Constructor

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

33.39.4 Properties

33.39.5 Arguments as Int64

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A 64-bit unsigned integer that contains the arguments for gestures that fit into 8 bytes.
Notes: (Read only property)

33.39.6 ArgumentsHigher as UInt32

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The higher 32bits from Arguments.
Notes: (Read only property)

33.39.7   ArgumentsLower as UInt32

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The lower 32bits from Arguments.
Notes: Most events just use lower bits.
(Read only property)

33.39.8   Flags as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The state of the gesture.
Notes: See kFlagBegin/kFlagInertia/kFlagEnd constants.
(Read only property)

33.39.9   ID as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identifier of the gesture command.
Notes: see kGestureID* constants.
(Read only property)

33.39.10   InstanceID as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: An internally used identifier for the structure.
Notes: (Read only property)

33.39.11   LocationInWindowX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A x coordinate associated with the gesture.
Notes: This is LocationX relative to target window.

(Read only property)

33.39.12 LocationInWindowY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A x coordinate associated with the gesture.
Notes: This is LocationY relative to target window.
(Read only property)

33.39.13 LocationX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A x coordinate associated with the gesture.
Notes: These coordinates are always relative to the origin of the screen.
(Read only property)

33.39.14 LocationY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A x coordinate associated with the gesture.
Notes: These coordinates are always relative to the origin of the screen.
(Read only property)

33.39.15 RotateAngle as Double

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The argument as angle for rotate gesture.
Notes: (Read only property)

33.39.16 SequenceID as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: An internally used identifier for the sequence.
Notes: (Read only property)

33.39.17   TargetWindow as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A handle to the window that is targeted by this gesture.
Notes: (Read only property)

33.39.18   Constants

Flags

Constant	Value	Description
kFlagBegin	1	A gesture is starting.
kFlagEnd	4	A gesture has finished.
kFlagInertia	2	A gesture has triggered inertia.

Gesture IDs

Constant	Value	Description
kGestureIDBegin	1	Begin of a gesture
kGestureIDEnd	2	End of a gesture
kGestureIDPan	4	Pan gesture
kGestureIDPressAndTap	7	Press and Tap gesture
kGestureIDRotate	5	Rotate gesture
kGestureIDTwoFingerTap	6	Two-Finger Tap
kGestureIDZoom	3	Zoom gesture

33.40   class WinMouseFilterMBS

33.40.1   class WinMouseFilterMBS

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to filter mouse events in the whole application.
Notes: You may be able to intercept all mouse actions and block mouse events.

You can only have one instance to receive events at a time.
Blog Entries

- News from the MBS Xojo Plugins Version 21.4

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.4

- MBS Xojo Plugins, version 21.4pr1

Xojo Developer Magazine

- 19.6, page 10: News

33.40.2   Methods

33.40.3   Constructor

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

33.40.4   Properties

33.40.5   Enabled as Boolean

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Global flag to enable/disable it.
Notes: Set true to enable or false to disable.
(Read and Write property)

33.40.6   IncludeMouseHOver as Boolean

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to include hover events.
Notes: By default false to disable hover events.
Set to true to include those.
(Read and Write property)

33.40.7 IncludeMouseMove as Boolean

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to include move events.
Notes: By default false to disable move events.
Set to true to include those.
(Read and Write property)

33.40.8 IncludeMouseWheel as Boolean

Plugin Version: 21.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether to include wheel events.
Notes: By default false to disable wheel events.
Set to true to include those.
(Read and Write property)

33.40.9 Events

33.40.10 ReceivedEvent(Message as Integer, mouseX as Integer, mouseY as Integer, windowHandle as Integer, HitTestCode as Integer, ExtraInfo as Integer) as Boolean

Plugin Version: 21.4, Platform: Windows, Targets: .

Function: Received an event.
Notes: Message: The identifier of the mouse message. See constants.
mouseX and mouseY: The x- and y-coordinates of the cursor, in screen coordinates.
WindowHandle: A handle to the window that will receive the mouse message corresponding to the mouse event.
HitTestCode: The hit-test value. For a list of hit-test values, see the description of the WM_NCHITTEST message.
ExtraInfo: Additional information associated with the message.

33.40.11 Constants

Mouse Events

Constant	Value	Description
kLButtonDoubleClick	&h0203	Left mouse button double click
kLButtonDown	&h0201	Left mouse button down
kLButtonUp	&h0202	Left mouse button up
kMButtonDoubleClick	&h0209	Middle mouse button double click
kMButtonDown	&h0207	Middle mouse button down
kMButtonUp	&h0208	Middle mouse button up
kMouseHOver	&h02a1	Mouse hover
kMouseHWheel	&h020e	Horizontal mouse wheel
kMouseLeave	&h02a3	Mouse Leave
kMouseMove	&h0200	Mouse moved
kMouseWheel	&h020a	Vertical mouse wheel
kRButtonDoubleClick	&h0206	Right mouse button double click
kRButtonDown	&h0204	Right mouse button down
kRButtonUp	&h0205	Right mouse button up
kXButtonDoubleClick	&h020d	Other mouse button double click
kXButtonDown	&h020b	Other mouse button down
kXButtonUp	&h020c	Other mouse button up

33.41   class WinPointerEventsMBS

33.41.1   class WinPointerEventsMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to capture touch events on Windows.
Notes: The plugin can only deliver events the Xojo runtime doesn't consume.
Blog Entries

- Windows Touch Events

- MBS Xojo / Real Studio Plugins, version 17.0pr4

Videos

- Presentation from Munich conference about MBS Plugins.

33.41.2   Methods

33.41.3   Close

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Cleans up.
Notes: Optional and does same as destructor, but now.
You can call this in a window's close event to shutdown the event handler class.

33.41.4   Constructor(control as DesktopUIControl)

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

Function: The constructor events to a control.
Notes: Call in a control's open event to attach event listener to the control.
See also:

33.41.5 Constructor(control as RectControl)

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: The constructor events to a control.
Notes: Call in a control's open event to attach event listener to the control.
See also:

33.41.6 Constructor(win as DesktopWindow)

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

Function: The constructor events to a window.
Notes: Call in a window's open event to attach event listener to the window.
See also:

33.41.7 Constructor(win as window)

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: The constructor events to a window.
Notes: Call in a window's open event to attach event listener to the window.
See also:

33.41.8 Constructor(WindowHandle as Integer)

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.
Notes: This method allows you to attach to any window.
So this works in console projects, too.
With any window, not just xojo windows.
See also:

33.41.9 EnableMouseInPointer(enable as boolean) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Enables the mouse to act as a pointer input device and send pointer events.
Notes: Requires Windows 8 or newer.

If the function succeeds, the return value is true.
If the function fails, the return value is false.
To get extended error information, call GetLastError.

This function can be called only once in the context of a process lifetime. Prior to the first call, Windows Store apps run with mouse-in-pointer enabled, as do any desktop applications that consume mshtml.dll. All other desktop applications run with mouse-in-pointer disabled.

On the first call in the process lifetime, the state is changed as specified and the call succeeds.
On subsequent calls, the state will not change. If the current state is not equal to the specified state, the call fails.

Call IsMouseInPointerEnabled to verify the mouse-in-pointer state.

33.41.10 GetGestureConfig(Control as DesktopUIControl, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer

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

Function: Retrieves the configuration for which Windows Touch gesture messages are sent from a window.
Notes: Same as other GetGestureConfig, but for a control, not a window.
See also:

- 33.41.11 GetGestureConfig(Control as RectControl, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1253

- 33.41.12 GetGestureConfig(win as DesktopWindow, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1253

- 33.41.13 GetGestureConfig(win as window, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1254

33.41.11 GetGestureConfig(Control as RectControl, config() as WinGesture-ConfigMBS, flags as Integer = 0) as Integer

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: Retrieves the configuration for which Windows Touch gesture messages are sent from a window.
Notes: Same as other GetGestureConfig, but for a control, not a window.
See also:

- 33.41.10 GetGestureConfig(Control as DesktopUIControl, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1252

- 33.41.12 GetGestureConfig(win as DesktopWindow, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1253

- 33.41.13 GetGestureConfig(win as window, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer 1254

33.41.12 GetGestureConfig(win as DesktopWindow, config() as WinGesture-ConfigMBS, flags as Integer = 0) as Integer

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

Function: Retrieves the configuration for which Windows Touch gesture messages are sent from a window.
Notes: win: The window to get the gesture configuration from.
Flags: A gesture command flag value indicating options for retrieving the gesture configuration. Can be 1 to return consolidated configuration for the specified window and its parent window chain.
Config: an array of gesture configuration structures that specify the gesture configuration.

Please pass an empty array and the plugin appends config objects.

Return function returns number of configs found.
Returns -1 in case of error.
See also:

33.41.13 GetGestureConfig(win as window, config() as WinGestureConfigMBS, flags as Integer = 0) as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: Retrieves the configuration for which Windows Touch gesture messages are sent from a window.
Notes: win: The window to get the gesture configuration from.
Flags: A gesture command flag value indicating options for retrieving the gesture configuration. Can be 1 to return consolidated configuration for the specified window and its parent window chain.
Config: an array of gesture configuration structures that specify the gesture configuration.

Please pass an empty array and the plugin appends config objects.

Return function returns number of configs found.
Returns -1 in case of error.
See also:

33.41.14 IsMouseInPointerEnabled as boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates whether EnableMouseInPointer is set for the mouse to act as a pointer input device and send pointer events.
Notes: Requires Windows 8 or newer.

If EnableMouseInPointer is set, the return value is true.
If EnableMouseInPointer is not set, the return value is false.

EnableMouseInPointer can be called only once in the context of a process lifetime. Prior to the first call, Windows Store apps run with mouse-in-pointer enabled, as do any desktop applications that consume mshtml.dll. All other desktop applications run with mouse-in-pointer disabled.

On the first call to EnableMouseInPointer in the process lifetime, the state is changed as specified and the call succeeds.

On subsequent calls to EnableMouseInPointer, the state will not change. If the current state is not equal to the specified state, the call fails.

Call IsMouseInPointerEnabled to verify the mouse-in-pointer state.

33.41.15 SetGestureConfig(Control as DesktopUIControl, config() as WinGestureConfigMBS = nil) as boolean

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

Function: Configures the messages that are sent from a window for Windows Touch gestures.
Notes: Same as other SetGestureConfig, but for a control, not a window.
See also:

- 33.41.16 SetGestureConfig(Control as RectControl, config() as WinGestureConfigMBS = nil) as boolean 1255

- 33.41.17 SetGestureConfig(win as DesktopWindow, config() as WinGestureConfigMBS = nil) as boolean 1256

- 33.41.18 SetGestureConfig(win as window, config() as WinGestureConfigMBS = nil) as boolean 1256

33.41.16 SetGestureConfig(Control as RectControl, config() as WinGestureConfigMBS = nil) as boolean

Plugin Version: 17.0, Platform: Windows, Targets: Desktop only.

Function: Configures the messages that are sent from a window for Windows Touch gestures.
Notes: Same as other SetGestureConfig, but for a control, not a window.
See also:

- 33.41.15 SetGestureConfig(Control as DesktopUIControl, config() as WinGestureConfigMBS = nil) as boolean 1255

- 33.41.17 SetGestureConfig(win as DesktopWindow, config() as WinGestureConfigMBS = nil) as boolean 1256

- 33.41.18 SetGestureConfig(win as window, config() as WinGestureConfigMBS = nil) as boolean 1256

33.41.17   SetGestureConfig(win as DesktopWindow, config() as WinGesture-ConfigMBS = nil) as boolean

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

Function: Configures the messages that are sent from a window for Windows Touch gestures.
Notes: Win: The window to set the gesture configuration on.
config: An array of gesture configuration structures that specify the gesture configuration.

If the function succeeds, the return value is true, else false.

If you don't expect to change the gesture configuration, call SetGestureConfig at window creation time. If you want to dynamically change the gesture configuration, call SetGestureConfig in response to GestureNotify event.
See also:

33.41.18   SetGestureConfig(win as window, config() as WinGestureConfigMBS = nil) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: Configures the messages that are sent from a window for Windows Touch gestures.
Notes: Win: The window to set the gesture configuration on.
config: An array of gesture configuration structures that specify the gesture configuration.

If the function succeeds, the return value is true, else false.

If you don't expect to change the gesture configuration, call SetGestureConfig at window creation time. If you want to dynamically change the gesture configuration, call SetGestureConfig in response to GestureNotify event.
See also:

33.41.19   Properties

33.41.20   WindowHandle as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The window handle.
Notes: (Read only property)

33.41.21   Events

33.41.22   Gesture(info as WinGestureInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Passes information about a gesture.
Notes: Return true, if you processed the message, else false.

33.41.23   GestureNotify

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Gives you a chance to set the gesture configuration.
Notes: When the GestureNotify event is received, the application can use SetGestureConfig to specify the gestures to receive.

33.41.24   PointerDeviceChange(Change as Integer, Param as Integer) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Sent to a window when there is a change in the settings of a monitor that has a digitizer attached to it.
Notes: This message contains information regarding the scaling of the display mode.
See kDeviceChange* constants.

33.41.25   PointerDeviceInRange(Param1 as Integer, Param2 as Integer) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Sent to a window when a pointer device is detected within range of an input digitizer.
Notes: This message contains information regarding the device and its proximity.

If the application processes this message, it should return true, else false.

33.41.26   PointerDeviceOutOfRange(Param1 as Integer, Param2 as Integer) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Sent to a window when a pointer device has departed the range of an input digitizer.
Notes: This message contains information regarding the device and its proximity.

If the application processes this message, it should return true, else false.

33.41.27   PointerDown(PointerID as Integer, Flags as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Posted when a pointer makes contact over the client area of a window.
Notes: This input message targets the window over which the pointer makes contact, and the pointer is implicitly captured to the window so that the window continues to receive input for the pointer until it breaks contact.

We provide pointer ID, flags, X and Y.

Note:
A hovering pointer has none of the button flags set. This is analogous to a mouse move with no mouse buttons down. An application can determine the buttons states of a hovering pen, for example, by calling GetPointerPenInfo and examining the flags that specify button states.

Note:
Because the pointer may make contact with the device over a non-trivial area, this point location may be a simplification of a more complex pointer area. Whenever possible, an application should use the complete pointer area information instead of the point location.

BitwiseAnd(flags, WinPointerInfoMBS.kFlagNew) <>0	A flag that indicates whether this message represents the first input generated by a new pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	A flag that indicates whether this message was generated by a pointer during its lifetime. This flag is not set on messages that indicate that the pointer has left detection range
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInContact) <>0	A flag that indicates whether this message was generated by a pointer that is in contact with the window surface. This flag is not set on messages that indicate a hovering pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagPrimary) <>0	Indicates that this pointer has been designated as primary.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFirstButton) <>0	A flag that indicates whether there is a primary action. This is analogous to a mouse left button down. A touch pointer will have this set when it is in contact with the digitizer surface. A pen pointer will have this set when it is in contact with the digitizer surface with no buttons pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagSecondButton) <>0	A flag that indicates whether there is a secondary action. This is analogous to a mouse right button down. A pen pointer will have this set when it is in contact with the digitizer surface with the pen barrel button pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagThirdButton) <>0	A flag that indicates whether there are one or more tertiary actions based on the pointer type; applications that wish to respond to tertiary actions must retrieve information specific to the pointer type to determine which tertiary buttons are pressed. For example, an application can determine the buttons states of a pen by calling GetPointerPenInfo and examining the flags that specify button states.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFourthButton) <>0	A flag that indicates whether the specified pointer took fourth action. Applications that wish to respond to fourth actions must retrieve information specific to the pointer type to determine if the first extended mouse (XButton1) button is pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFifthButton) <>0	A flag that indicates whether the specified pointer took fifth action. Applications that wish to respond to fifth actions must retrieve information specific to the pointer type to determine if the second extended mouse (XButton2) button is pressed.

If an application processes this message, it should return true, else false.

Important When a window loses capture of a pointer and it receives the PointerCaptureChanged notification, it typically will not receive any further notifications. For this reason, it is important that you not make any assumptions based on evenly paired PointerDown/PointerUp or PointerEnter/PointerLeave notifications.

Each pointer has a unique pointer identifier during its lifetime. The lifetime of a pointer begins when it is first detected.
A PointerEnter event is generated if a hovering pointer is detected. A PointerDown event followed by a PointerEnter event is generated if a non-hovering pointer is detected.
During its lifetime, a pointer may generate a series of PointerUpdate events while it is hovering or in contact. The lifetime of a pointer ends when it is no longer detected. This generates a PointerLeave event.
When a pointer is aborted, kFlagCanceled is set.
A WM_POINTERLEAVE message may also be generated when a non-captured pointer moves outside the bounds of a window.

To convert the lParam parameter to a POINTS structure, use the MAKEPOINTS macro.
To retrieve further information associated with the message, use the GetPointerInfo function.
To determine the keyboard modifier key states associated with this message, use the GetKeyState function. For example, to detect that the ALT key was pressed, check whether GetKeyState(VK_MENU) <0.
Note that if the application does not process this message, DefWindowProc may generate one or more WM_GESTURE messages if the sequence of input from this and, possibly, other pointers is recognized as

a gesture. If a gesture is not recognized, DefWindowProc may generate mouse input.

If an application selectively consumes some pointer input and passes the rest to DefWindowProc, the resulting behavior is undefined.

When a window loses capture of a pointer and receives the WM_POINTERCAPTURECHANGED notification, it will typically not receive any further notifications. Therefore, it is important that a window does not make any assumptions of its pointer status, regardless of whether it receives evenly paired DOWN / UP or ENTER / LEAVE notifications.

33.41.28 PointerEnter(PointerID as Integer, Flags as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Sent to a window when a new pointer enters detection range over the window (hover) or when an existing pointer moves within the boundaries of the window.
Notes: We provide Pointer ID, Flags, X and Y.

BitwiseAnd(flags, WinPointerInfoMBS.kFlagNew) <>0	Indicates whether this message is the first message generated by a new pointer entering detection range (hover).
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	Indicates whether this message was generated by a pointer that has not left detection range. This flag is always set for PointerEnter messages.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInContact) <>0	A flag that indicates whether this message was generated by a pointer that is in contact. This flag is not set for a pointer in detection range (hover).

Contains the point location of the pointer.

Note:

Because the pointer may make contact with the device over a non-trivial area, this point location may be a simplification of a more complex pointer area. Whenever possible, an application should use the complete pointer area information instead of the point location.

If an application processes this message, it should return true, else false.

The PointerEnter notification can be used by a window to provide feedback to the user while the pointer is over its surface or to otherwise react to the presence of a pointer over its surface.

This notification is only sent to the window that is receiving input for the pointer. The following table lists some of the situations in which this notification is sent.

Action	Flags Set	Notifications Sent To
A new pointer enters detection range (hover).	kFlagNew and kFlagInRange	Window over which the pointer enters detection range.
A hovering pointer crosses within the window boundaries.	kFlagInRange	Window within which the pointer has crossed.

Important:

When a window loses capture of a pointer and it receives the PointerCaptureChange notification, it typically will not receive any further notifications. For this reason, it is important that you not make any assumptions

based on evenly paired PointerDown/PointerUp or PointerEnter/PointerLeave notifications.

When inputs come from the mouse, as a result of mouse and pointer message integration, PointerEnter is not sent.

33.41.29 PointerHWheel(PointerID as Integer, Delta as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Posted to the window with foreground keyboard focus when a horizontal scroll wheel is rotated.
Notes: We provide, the pointer ID, the delta and x/y coordinate.
Info object provides pointer details.

To retrieve the wheel scroll units, use the inputData filed of the WinPointerInfoMBS object. This field contains a signed value and is expressed in a multiple of WHEEL_DELTA (120). A positive value indicates a rotation forward and a negative value indicates a rotation backward.
Note that the wheel inputs may be delivered even if the mouse cursor is located outside of application‚Äôs window. The wheel messages are delivered in a way very similar to the keyboard inputs. The focus window of the foregournd message queue receives the wheel messages.

33.41.30 PointerLeave(PointerID as Integer, Flags as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Sent to a window when a pointer leaves detection range over the window (hover) or when a pointer moves outside the boundaries of the window.
Notes: We provide the pointer ID, Flags, X and Y values.

BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	Indicates whether this message was generated by a pointer that has not left detection range. This flag is not set when the pointer leaves the detection range of the window.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInContact) <>0	A flag that indicates whether this message was generated by a pointer that is in contact. This flag is not set for a pointer in detection range (hover).

Note:
Because the pointer may make contact with the device over a non-trivial area, this point location may be a simplification of a more complex pointer area. Whenever possible, an application should use the complete pointer area information instead of the point location.

If an application processes this message, it should return true, else false.

The PointerLeave notification can be used by a window to change mode or stop any feedback to the user while the pointer is over the window surface.

This notification is only sent to the window that is receiving input for the pointer. The following table lists some of the situations in which this notification is sent.

Action	Flags Set	Notifications Sent To
A hovering pointer crosses window boundaries.	BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	Window outside of whose boundary the pointer moved.
A pointer goes out of detection range.	N/A	Window for which the pointer leaves detection range.

Important:
When a window loses capture of a pointer and it receives the PointerCaptureChange notification, it typically will not receive any further notifications. For this reason, it is important that you not make any assumptions based on evenly paired PointerDown/PointerUp or PointerEnter/PointerLeave notifications.

If contact is maintained with the input digitizer and the pointer moves outside the window, PointerLeave is not generated. PointerLeave is generated only when a hovering pointer crosses window boundaries or contact is terminated.

PointerLeave is posted to the posted message queue if the input is originated from a mouse device.

33.41.31   PointerUp(PointerID as Integer, Flags as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Posted when a pointer that made contact over the client area of a window breaks contact.
Notes: This input message targets the window over which the pointer makes contact and the pointer is, at that point, implicitly captured to the window so that the window continues to receive input messages including the WM_POINTERUP notification for the pointer until it breaks contact.

We provide pointer ID, flags, X and Y.

See Pointer Flags for more details.

Note:
A hovering pointer has none of the button flags set. This is analogous to a mouse move with no mouse buttons down. An application can determine the buttons states of a hovering pen, for example, by calling GetPointerPenInfo and examining the flags that specify button states.

Note:
Because the pointer may make contact with the device over a non-trivial area, this point location may be a simplification of a more complex pointer area. Whenever possible, an application should use the complete

BitwiseAnd(flags, WinPointerInfoMBS.kFlagNew) <>0	A flag that indicates whether this message represents the first input generated by a new pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	A flag that indicates whether this message was generated by a pointer during its lifetime. This flag is not set on messages that indicate that the pointer has left detection range
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInContact) <>0	A flag that indicates whether this message was generated by a pointer that is in contact with the window surface. This flag is not set on messages that indicate a hovering pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagPrimary) <>0	Indicates that this pointer has been designated as primary.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFirstButton) <>0	A flag that indicates whether there is a primary action. This is analogous to a mouse left button down. A touch pointer will have this set when it is in contact with the digitizer surface. A pen pointer will have this set when it is in contact with the digitizer surface with no buttons pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagSecondButton) <>0	A flag that indicates whether there is a secondary action. This is analogous to a mouse right button down. A pen pointer will have this set when it is in contact with the digitizer surface with the pen barrel button pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagThirdButton) <>0	A flag that indicates whether there are one or more tertiary actions based on the pointer type; applications that wish to respond to tertiary actions must retrieve information specific to the pointer type to determine which tertiary buttons are pressed. For example, an application can determine the buttons states of a pen by calling GetPointerPenInfo and examining the flags that specify button states.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFourthButton) <>0	A flag that indicates whether the specified pointer took fourth action. Applications that wish to respond to fourth actions must retrieve information specific to the pointer type to determine if the first extended mouse (XButton1) button is pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFifthButton) <>0	A flag that indicates whether the specified pointer took fifth action. Applications that wish to respond to fifth actions must retrieve information specific to the pointer type to determine if the second extended mouse (XButton2) button is pressed.

pointer area information instead of the point location.

If an application processes this message, it should return true, else false.

Each pointer has a unique pointer identifier during its lifetime. The lifetime of a pointer begins when it is first detected.
A PointerEnter event is generated if a hovering pointer is detected. A PointerDown event followed by a PointerEnter event is generated if a non-hovering pointer is detected.
During its lifetime, a pointer may generate a series of PointerUpdate event while it is hovering or in contact. The lifetime of a pointer ends when it is no longer detected. This generates a WM_POINTERLEAVE message.
When a pointer is aborted, kFlagCanceled is set.
A PointerLeave event may also be generated when a non-captured pointer moves outside the bounds of a window.
To obtain the horizontal and vertical position of a pointer, use the following:

If the application does not process this message, Windows may generate one or more Gesture event if the sequence of input from this and, possibly, other pointers is recognized as a gesture. If a gesture is not recognized, DefWindowProc may generate mouse input.
If an application selectively consumes some pointer input and passes the rest back to Windows, the resulting behavior is undefined.

33.41.32   PointerUpdate(PointerID as Integer, Flags as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Posted to provide an update on a pointer that made contact over the client area of a window or on a hovering uncaptured pointer over the client area of a window.
Notes: While the pointer is hovering, the message targets whichever window the pointer happens to be over. While the pointer is in contact with the surface, the pointer is implicitly captured to the window over which the pointer made contact and that window continues to receive input for the pointer until it breaks contact.

We provide pointer ID, flags, X and Y.

BitwiseAnd(flags, WinPointerInfoMBS.kFlagNew) <>0	A flag that indicates whether this message represents the first input generated by a new pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInRange) <>0	A flag that indicates whether this message was generated by a pointer during its lifetime. This flag is not set on messages that indicate that the pointer has left detection range
BitwiseAnd(flags, WinPointerInfoMBS.kFlagInContact) <>0	A flag that indicates whether this message was generated by a pointer that is in contact with the window surface. This flag is not set on messages that indicate a hovering pointer.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagPrimary) <>0	Indicates that this pointer has been designated as primary.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFirstButton) <>0	A flag that indicates whether there is a primary action. This is analogous to a mouse left button down. A touch pointer will have this set when it is in contact with the digitizer surface. A pen pointer will have this set when it is in contact with the digitizer surface with no buttons pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagSecondButton) <>0	A flag that indicates whether there is a secondary action. This is analogous to a mouse right button down.

A pen pointer will have this set when it is in contact with the digitizer surface with the pen barrel button pressed.

BitwiseAnd(flags, WinPointerInfoMBS.kFlagThirdButton) <>0	A flag that indicates whether there are one or more tertiary actions based on the pointer type; applications that wish to respond to tertiary actions must retrieve information specific to the pointer type to determine which tertiary buttons are pressed. For example, an application can determine the buttons states of a pen by calling GetPointerPenInfo and examining the flags that specify button states.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFourthButton) <>0	A flag that indicates whether the specified pointer took fourth action. Applications that wish to respond to fourth actions must retrieve information specific to the pointer type to determine if the first extended mouse (XButton1) button is pressed.
BitwiseAnd(flags, WinPointerInfoMBS.kFlagFifthButton) <>0	A flag that indicates whether the specified pointer took fifth action. Applications that wish to respond to fifth actions must retrieve information specific to the pointer type to determine if the second extended mouse (XButton2) button is pressed.

See Pointer Flags for more details.

Note:
A hovering pointer has none of the button flags set. This is analogous to a mouse move with no mouse

buttons down. An application can determine the buttons states of a hovering pen, for example, by calling GetPointerPenInfo and examining the flags that specify button states.

Note:
Because the pointer may make contact with the device over a non-trivial area, this point location may be a simplification of a more complex pointer area. Whenever possible, an application should use the complete pointer area information instead of the point location.

If an application processes this message, it should return true, else false.

Each pointer has a unique pointer identifier during its lifetime. The lifetime of a pointer begins when it is first detected.
A PointerEnterevent is generated if a hovering pointer is detected. A PointerDown event followed by a PointerEnter event is generated if a non-hovering pointer is detected.
During its lifetime, a pointer may generate a series of PointerUpdate events while it is hovering or in contact. The lifetime of a pointer ends when it is no longer detected. This generates a PointerLeave event.
When a pointer is aborted, kFlagCanceled is set.
A PointerLeave event may also be generated when a non-captured pointer moves outside the bounds of a window.

If the application does not process this message, Windows may generate one or more Gesture event if the sequence of input from this and, possibly, other pointers is recognized as a gesture. If a gesture is not recognized, Windows may generate mouse input.
If an application selectively consumes some pointer input and passes the rest back to Windows, the resulting behavior is undefined.
If the application does not process these messages as fast as they are generated, some moves may be coalesced. The history of inputs that were coalesced into this message can be retrieved using the GetPointerInfoHistory function.

33.41.33   PointerWheel(PointerID as Integer, Delta as Integer, X as Integer, Y as Integer, info as WinPointerInfoMBS) as boolean

Plugin Version: 16.2, Platform: Windows, Targets: .

Function: Posted to the window with foreground keyboard focus when a vertical scroll wheel is rotated.
Notes: We provide, the pointer ID, the delta and x/y coordinate.
Info object provides pointer details.

To retrieve the wheel scroll units, use the inputData filed of the WinPointerInfoMBS object. This field contains a signed value and is expressed in a multiple of WHEEL_DELTA (120). A positive value indicates a rotation forward and a negative value indicates a rotation backward.
Note that the wheel inputs may be delivered even if the mouse cursor is located outside of application‚Äôs window. The wheel messages are delivered in a way very similar to the keyboard inputs. The focus window

of the foregournd message queue receives the wheel messages.

33.41.34   Constants

Device Change Flags

Constant	Value	Description
kDeviceChangeArrival	1	A new device is attached.
kDeviceChangeAspectRatioPreserved	2048	The display aspect ratio.
kDeviceChangeMapping	256	The change in display to digitizer mapping.
kDeviceChangeModeCentered	128	Centered display mode.
kDeviceChangeModeDefault	64	The default display mode.
kDeviceChangeOrientation0	4	Orientation of the device: $0\neg\infty$
kDeviceChangeOrientation180	16	Orientation of the device: $180\neg\infty$
kDeviceChangeOrientation270	32	Orientation of the device: $270\neg\infty$
kDeviceChangeOrientation90	8	Orientation of the device: $90\neg\infty$
kDeviceChangeOrigin	1024	The display origin.
kDeviceChangeRemoval	2	A device has been detached.
kDeviceChangeResolution	512	Display resolution.

33.42 class WinPointerInfoMBS

33.42.1 class WinPointerInfoMBS

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for pointer input details.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.
Blog Entries

- Windows Touch Events

33.42.2 Methods

33.42.3 Constructor

Plugin Version: 16.2, Platform: Windows, Targets: Desktop only.

Function: The private constructor.

33.42.4 Properties

33.42.5 ButtonChangeType as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Specifies the change in button state between this input and the previous input.
Notes: See kPointerChange* constants.
(Read only property)

33.42.6 frameId as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: An identifier common to multiple pointers for which the source device reported an update in a single input frame.
Notes: For example, a parallel-mode multi-touch digitizer may report the positions of multiple touch contacts in a single update to the system.
Note that frame identifier is assigned as input is reported to the system for all pointers across all devices. Therefore, this field may not contain strictly sequential values in a single series of messages that a window receives. However, this field will contain the same numerical value for all input updates that were reported in the same input frame by a single device.

(Read only property)

33.42.7   HimetricLocationRawX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The screen coordinates of the pointer, in HIMETRIC units.
Notes: For adjusted screen coordinates, see HimetricLocation.
(Read only property)

33.42.8   HimetricLocationRawY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The screen coordinates of the pointer, in HIMETRIC units.
Notes: For adjusted screen coordinates, see HimetricLocation.
(Read only property)

33.42.9   HimetricLocationX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The predicted screen coordinates of the pointer, in HIMETRIC units.
Notes: The predicted value is based on the pointer position reported by the digitizer and the motion of the pointer. This correction can compensate for visual lag due to inherent delays in sensing and processing the pointer location on the digitizer. This is applicable to pointers of type touch. For other pointer types, the predicted value will be the same as the non-predicted value (see ptHimetricLocationRaw).
(Read only property)

33.42.10   HimetricLocationY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The predicted screen coordinates of the pointer, in HIMETRIC units.
Notes: The predicted value is based on the pointer position reported by the digitizer and the motion of the pointer. This correction can compensate for visual lag due to inherent delays in sensing and processing the pointer location on the digitizer. This is applicable to pointers of type touch. For other pointer types, the predicted value will be the same as the non-predicted value (see ptHimetricLocationRaw).
(Read only property)

33.42.11 historyCount as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Count of inputs that were coalesced into this message.
Notes: This count matches the total count of entries that can be returned by a call to GetPointerInfoHistory. If no coalescing occurred, this count is 1 for the single input represented by the message.
(Read only property)

33.42.12 hwndTarget as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Window to which this message was targeted.
Notes: If the pointer is captured, either implicitly by virtue of having made contact over this window or explicitly using the pointer capture API, this is the capture window. If the pointer is uncaptured, this is the window over which the pointer was when this message was generated.
(Read only property)

33.42.13 InputData as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: A value whose meaning depends on the nature of input.
Notes: When flags indicate kFlagWheel, this value indicates the distance the wheel is rotated, expressed in multiples or factors of WHEEL_DELTA. A positive value indicates that the wheel was rotated forward and a negative value indicates that the wheel was rotated backward.
When flags indicate kFlagWheel, this value indicates the distance the wheel is rotated, expressed in multiples or factors of WHEEL_DELTA. A positive value indicates that the wheel was rotated to the right and a negative value indicates that the wheel was rotated to the left.
(Read only property)

33.42.14 KeyStates as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates which keyboard modifier keys were pressed at the time the input was generated.
Notes: May be zero or a combination of the following values:

(Read only property)

8   A SHIFT key was pressed.

4   A CTRL key was pressed.

33.42.15   PerformanceCount as Int64

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The value of the high-resolution performance counter when the pointer message was received (high-precision, 64 bit alternative to Time).
Notes: The value can be calibrated when the touch digitizer hardware supports the scan timestamp information in its input report.
(Read only property)

33.42.16   PixelLocationRawX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The screen coordinates of the pointer, in pixels.
Notes: For adjusted screen coordinates, see PixelLocation.
(Read only property)

33.42.17   PixelLocationRawY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The screen coordinates of the pointer, in pixels.
Notes: For adjusted screen coordinates, see PixelLocation.
(Read only property)

33.42.18   PixelLocationX as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The predicted screen coordinates of the pointer, in pixels.
Notes: The predicted value is based on the pointer position reported by the digitizer and the motion of the pointer. This correction can compensate for visual lag due to inherent delays in sensing and processing the pointer location on the digitizer. This is applicable to pointers of type touch. For other pointer types, the predicted value will be the same as the non-predicted value (see ptPixelLocationRaw).
(Read only property)

33.42.19 PixelLocationY as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The predicted screen coordinates of the pointer, in pixels.
Notes: The predicted value is based on the pointer position reported by the digitizer and the motion of the pointer. This correction can compensate for visual lag due to inherent delays in sensing and processing the pointer location on the digitizer. This is applicable to pointers of type touch. For other pointer types, the predicted value will be the same as the non-predicted value (see ptPixelLocationRaw).
(Read only property)

33.42.20 pointerFlags as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: May be any reasonable combination of flags from the flags constants.
Notes: See kFlag* constants.
(Read only property)

33.42.21 pointerId as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: An identifier that uniquely identifies a pointer during its lifetime.
Notes: A pointer comes into existence when it is first detected and ends its existence when it goes out of detection range. Note that if a physical entity (finger or pen) goes out of detection range and then returns to be detected again, it is treated as a new pointer and may be assigned a new pointer identifier.
(Read only property)

33.42.22 pointerType as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The pointer type.
Notes: See kType* constants.
(Read only property)

33.42.23 sourceDevice as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Handle to the source device that can be used in calls to the raw input device API and the digitizer device API.
Notes: (Read only property)

33.42.24   Time as Integer

Plugin Version: 16.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: The time of the event.
Notes: Value is 0 or the time stamp of the message, based on the system tick count when the message was received.
The application can specify the input time stamp in either Time or PerformanceCount. The value cannot be more recent than the current tick count or QueryPerformanceCount (QPC) value of the injection thread. Once a frame is injected with a time stamp, all subsequent frames must include a timestamp until all contacts in the frame go to an UP state. The custom timestamp value must also be provided for the first element in the contacts array. The time stamp values after the first element are ignored. The custom timestamp value must increment in every injection frame.
When PerformanceCount is specified, the time stamp will be converted to the current time in .1 millisecond resolution upon actual injection. If a custom PerformanceCount resulted in the same .1 millisecond window from the previous injection, ERROR_NOT_READY is returned and injection will not occur. While injection will not be invalidated immediately by the error, the next successful injection must have a PerformanceCount value that is at least 0.1 millisecond from the previously successful injection. This is also true if Time is used.
If both Time and PerformanceCount are specified in InjectTouchInput, ERROR_INVALID_PARAMETER is returned.
InjectTouchInput cannot switch between Time and PerformanceCount once injection has started.
If neither Time and PerformanceCount are specified, InjectTouchInput allocates the timestamp based on the timing of the call. If InjectTouchInput calls are repeatedly less than 0.1 millisecond apart, ERROR_NOT_READY might be returned. The error will not invalidate the input immediately, but the injection application needs to retry the same frame again for injection to succeed.
(Read only property)

33.42.25   Constants

Constants

Constant	Value	Description
kPointerChangeFifthButtonDown	9	The fifth button transitioned to a pressed state.
kPointerChangeFifthButtonUp	10	The fifth button transitioned to a released state.
kPointerChangeFirstButtonDown	1	The first button transitioned to a pressed state.
kPointerChangeFirstButtonUp	2	The first button transitioned to a released state.
kPointerChangeFourthButtonDown	7	The fourth button transitioned to a pressed state.
kPointerChangeFourthButtonUp	8	The fourth button transitioned to a released state.
kPointerChangeNone	0	No change in button state.
kPointerChangeSecondButtonDown	3	The second button transitioned to a pressed state.
kPointerChangeSecondButtonUp	4	The second button transitioned to a released state.
kPointerChangeThirdButtonDown	5	The third button transitioned to a pressed state.
kPointerChangeThirdButtonUp	6	The third button transitioned to a released state.

Flag Constants

Constant	Value	Description
kFlagCanceled	&h8000	Indicates that the pointer is departing in an abnormal manner, such as when the system receives invalid input for the pointer or when a device with active pointers departs abruptly. If the application receiving the input is in a position to do so, it should treat the interaction as not completed and reverse any effects of the concerned pointer.
kFlagCaptureChanged	&h200000	Indicates that this pointer was captured by (associated with) another element and the original element has lost capture.
kFlagConfidence	&h4000	Confidence is a suggestion from the source device about whether the pointer represents an intended or accidental interaction, which is especially relevant for touch pointers where an accidental interaction (such as with the palm of the hand) can trigger input. The presence of this flag indicates that the source device has high confidence that this input is part of an intended interaction.
kFlagDown	&h10000	Indicates that this pointer transitioned to a down state; that is, it made contact with the digitizer surface.
kFlagFifthButton	&h100	Analogous to a second extended mouse (XButton2) button down. A touch pointer does not use this flag. A pen pointer does not use this flag. A mouse pointer has this flag set when the second extended mouse (XBUTTON2) button is down.
kFlagFirstButton	&h10	Indicates a primary action, analogous to a left mouse button down. A touch pointer has this flag set when it is in contact with the digitizer surface. A pen pointer has this flag set when it is in contact with the digitizer surface with no buttons pressed. A mouse pointer has this flag set when the left mouse button is down.
kFlagFourthButton	&h80	Analogous to a first extended mouse (XButton1) button down. A touch pointer does not use this flag. A pen pointer does not use this flag. A mouse pointer has this flag set when the first extended mouse (XBUTTON1) button is down.
kFlagHasTransform	&h400000	Indicates that this pointer has an associated transform.
kFlagHWheel	&h100000	Indicates input associated with a pointer h-wheel. For mouse pointers, this is equivalent to the action of the mouse horizontal scroll wheel.
kFlagInContact	4	Indicates that this pointer is in contact with the digitizer surface. When this flag is not set, it indicates a hovering pointer.
kFlagInRange	2	Indicates that this pointer continues to exist. When this flag is not set, it indicates the pointer has left detection range. This flag is typically not set only when a hovering pointer leaves detection range (kFlagUpdate is set) or when a pointer in contact with a window surface leaves detection range (kFlagUp is set).
kFlagNew	1	Indicates the arrival of a new pointer.
kFlagNone	0	Default
kFlagPrimary	&h2000	Indicates that this pointer has been designated as the primary pointer. A primary pointer is a single pointer that can perform actions beyond those available to non-primary pointers. For example, when a primary pointer makes contact with a window‚Äôs surface, it may provide the window an opportunity to activate by sending it a PointerActivate event. The primary pointer is identified from all current user interactions on the system (mouse, touch, pen, and so on). As such, the primary pointer might not be associated with your app. The first contact in a multi-touch interaction is set as the primary pointer. Once a primary pointer is identified, all contacts must be lifted before a new contact can be identified as a primary pointer. For apps that don't process pointer input, only the primary pointer's events are promoted to mouse events.
kFlagSecondButton	&h20	Indicates a secondary action, analogous to a right mouse button down. A touch pointer does not use this flag. A pen pointer has this flag set when it is in contact with the digitizer surface with the pen barrel button pressed. A mouse pointer has this flag set when the right mouse button is down.

Pointer Types

Constant	Value	Description
kTypeMouse	4	Mouse pointer type.
kTypePen	3	Pen pointer type.
kTypePointer	1	Generic pointer type.
		This type never appears in pointer messages or pointer data. Some data query functions allow the caller to restrict the query to specific pointer type. The kTypePointer type can be used in these functions to specify that the query is to include pointers of all types
kTypeTouch	2	Touch pointer type.
kTypeTouchpad	5	Touchpad pointer type (Windows 8.1 and later).

33.43   control WinPreviewControlMBS

33.43.1   control WinPreviewControlMBS

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The preview control for Windows.
Notes: Used to show PDF documents on Windows using the built-in preview handler on Windows 10 and newer.
Can also be used to use other built-in preview handlers on Windows by setting classID.

Since the control is handled by Windows, the key and mouse events are not expected to work.

Available on Windows 10 and newer.
Use QLPreviewViewControlMBS or PDFViewControlMBS for macOS to show previews and QLPreview-ControllerMBS or PDFViewIOSControlMBS for iOS.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3

- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.3

- PDF Viewer controls for iOS

- PDF Viewer control for Windows

- MBS Xojo Plugins, version 23.3pr2

Xojo Developer Magazine

- 21.5, page 10: News

33.43.2   Methods

33.43.3   LoadData(data as MemoryBlock)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.
See also:

- 33.43.4 LoadData(data as string)   1277

33.43.4 LoadData(data as string)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given data.
Notes: Lasterror is set.
If loading from data is not supported for preview, we use a temporary file instead.
See also:

33.43.5 LoadFile(file as folderitem)

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: Loads the given file.
Notes: Lasterror is set.
If the preview handler does not support reading file, the plugin reads the file and passes as data.

If the file path is invalid, the preview will show a file not found error message.

33.43.6 Properties

33.43.7 classID as String

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The class ID for the previewer to use.
Notes: If empty, uses PDF viewer.
Available preview classes in a typical Windows 10 installation:

Class ID	Description
{ 13D3C4B8-B179-4ebb-BF62-F704173E7448 }	Windows Contact Preview Handler
{ 1531d583-8375-4d3f-b5fb-d23bbd169f22 }	Windows TXT Previewer
{ 53BEDF0B-4E5B-4183-8DC9-B844344FA104 }	Microsoft Windows MAPI Preview Handler
{ 8a7cae0e-5951-49cb-bf20-ab3fa1e44b01 }	Windows Font previewer
{ 92dbad9f-5025-49b0-9078-2d78f935e341 }	Microsoft Windows Mail Mime Preview Handler
{ a42c2ccb-67d3-46fa-abe6-7d2f3488c7a3 }	Windows RTF Previewer
{ b9815375-5d7f-4ce2-9245-c9d4da436930 }	Microsoft Windows Mail Mime Preview Handler
{ BFD468D2-D0A0-4bdc-878C-E69C2F5B435D }	Microsoft Windows Mail Html Preview Handler
{ E64164EB-1AE0-4C50-BAEF-A413C2B3A4BC }	Microsoft 3MF Shell Thumbnail and Preview Handler
{ f8b8412b-dea3-4130-b36c-5e8be73106ac }	Microsoft Windows Mail Html Preview Handler
{ 3A84F9C2-6164-485C-A7D9-4B27F8AC009E }	Microsoft PDF Previewer
{ DC6EFB56-9CFA-464D-8880-44885D7DC193 }	Adobe PDF Preview Handler for Vista

(Read and Write property)

33.43.8   Lasterror as Integer

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The last error code.
Notes: (Read and Write property)

33.43.9   LasterrorString as String

Plugin Version: 23.3, Platform: Windows, Targets: Desktop only.

Function: The error message for the error number in the lasterror property.
Notes: (Read only property)

33.43.10   Events

33.43.11   Close

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The close event.
In Xojo version 2021r3 and newer this event is named Closing.

33.43.12   Configure

Plugin Version: 23.3, Platform: Windows, Targets: .

Function: The event called while initializing, where you can apply properties just before plugin reads them.
Notes: e.g. set classID here in code.

33.43.13   ConstructContextualMenu(base as MenuItem, x as Integer, y as Integer) as Boolean

Plugin Version: 23.3, Platform: Windows, Targets: .

Function: This event is called when it is appropriate to display a contextual menu for the control.

33.43.14 ContextualMenuAction(hitItem as MenuItem) as Boolean

Plugin Version: 23.3, Platform: Windows, 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.

33.43.15 EnableMenuItems

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The enable menu items event.
In Xojo version 2021r3 and newer this event is named MenuBarSelected.

33.43.16 GotFocus

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The got focus event.
In Xojo version 2021r3 and newer this event is named FocusReceived.

33.43.17 LostFocus

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The lost focus event.
In Xojo version 2021r3 and newer this event is named FocusLost.

33.43.18 Open

Plugin Version: 23.3, Platform: Windows, Targets: .

Function:

The open event.
In Xojo version 2021r3 and newer this event is named Opening.

33.44 class WinThreadPoolMBS

33.44.1 class WinThreadPoolMBS

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class to control Windows Thread Pool.
Example:

static pool as WinThreadPoolMBS // global or window property

pool = New WinThreadPoolMBS
pool.ThreadMaximum = 4

Notes: You can have an instance of this class to create and define a thread pool.
If you never initialize a pool with this class, the default pool is used.
Blog Entries

- News from the MBS Xojo Plugins in version 21.5

- MonkeyBread Software Releases the MBS Xojo Plugins in version 21.5

- Thread Pool for Windows

- MBS Xojo Plugins, version 21.5pr4

33.44.2 Methods

33.44.3 Constructor

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The constructor.

33.44.4 Destructor

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.

33.44.5   Properties

33.44.6   Handle as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal handle of the thread pool.
Notes: (Read and Write property)

33.44.7   LastError as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code.
Notes: (Read and Write property)

33.44.8   StackCommit as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The stack commit size.
Notes: Default is 4 KB for the default stack grow size.
The rate the stack grows.
Should be in page size multiplies, e.g. 4096.

Sets the stack reserve and commit sizes for new threads in the specified thread pool. Stack reserve and commit sizes for existing threads are not changed.
(Read and Write property)

33.44.9   StackReserve as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The stack reservie size.
Example:

static pool as WinThreadPoolMBS // property or global

// increase stack size from default
pool = New WinThreadPoolMBS
pool.StackReserve = 2048*1024

Notes: Default is 1 MB for the default stack size.
The rate the stack grows.
Should be in page size multiplies, e.g. 4096.

Sets the stack reserve and commit sizes for new threads in the specified thread pool. Stack reserve and commit sizes for existing threads are not changed.
(Read and Write property)

33.44.10   ThreadMaximum as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The upper limit for thread count.
Example:

static pool as WinThreadPoolMBS // property or global

pool = New WinThreadPoolMBS
pool.StackReserve = 2048*1024
pool.ThreadMaximum = 4

Notes: Default for Windows is 500, but it may be much lower in practice as it's relative to the number of CPU cores and jobs.
(Read and Write property)

33.44.11   ThreadMinimum as Integer

Plugin Version: 21.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The lower limit for thread count.
Notes: Default 0 to allow having no thread.
(Read and Write property)

33.45   class WMIObjectMBS

33.45.1   class WMIObjectMBS

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a WMI object.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 13.5pr6
- MBS Xojo / Real Studio Plugins, version 13.4pr2
- MBS Xojo / Real Studio Plugins, version 13.4pr1
- MonkeyBread Software Releases the MBS Real Studio plug-ins in version 13.1
- MBS Real Studio Plugins, version 13.1pr15
- MBS Releases the MBS Real Studio plug-ins in version 12.0
- MBS Real Studio Plugins, version 11.4pr1

33.45.2   Methods

33.45.3   GetNames as string()

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Queries the names of all properties in this object.

33.45.4   GetProperty(Name as string) as Variant

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a property as a variant.
Notes: Objects, booleans, strings and numbers are supported directly. Also string arrays in version 13.1.
Other values are casted to string first.

33.45.5   GetPropertyBoolean(Name as string) as Boolean

Plugin Version: 13.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a value as an integer.
Notes: Value must be of a numeric type.

33.45.6 GetPropertyDouble(Name as string) as Double

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a value as a double.
Notes: Value must be of a numeric type.

33.45.7 GetPropertyInt64(Name as string) as Int64

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a property as an integer value.

33.45.8 GetPropertyInteger(Name as string) as Integer

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a value as an integer.
Example:

dim w as WindowsWMIMBS // your WMI object

MsgBox str(w.GetPropertyInteger("MaxClockSpeed")) // uint32

Notes: Value must be of a numeric type.

33.45.9 GetPropertyObject(Name as string) as WMIObjectMBS

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a value as an object.
Notes: Value must be of an IUnknown type.

33.45.10 GetPropertyString(Name as string) as string

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a property as a string.
Example:

dim w as WindowsWMIMBS
MsgBox w.GetPropertyString("Name") // string

33.45.11   GetPropertyStringArray(Name as string) as string()

Plugin Version: 13.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets a property as a string array.
Example:

dim w as WindowsWMIMBS
MsgBox join(w.GetPropertyStringArray("Name"), ", ")

33.45.12   GetPropertyType(Name as string) as Integer

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the type of a property.
Notes: You can send in the value you get here and the plugin can be changed to handle this type, too. Currently only numbers and strings are handled.

33.45.13   GetPropertyTypeString(Name as string) as string

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gets the type of a property as a string.
Notes: For example type 8 is returned as "string".

33.45.14   Properties

33.45.15   Handle as Integer

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

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

33.45.16 Lasterror as Integer

Plugin Version: 12.0, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code reported from one of the functions.
Notes: -1 is the value used by the plugin if the function is not available.
(Read and Write property)

33.45.17 LasterrorMessage as String

Plugin Version: 13.5, Platform: Windows, Targets: Desktop, Console & Web.

Function: The error message for the last error.
Notes: This may help to debug the errors.
(Read only property)

Chapter 34

Windows Console

34.1 class ConsoleStateMBS

34.1.1 class ConsoleStateMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for handling a console window state.

34.1.2 Properties

34.1.3 BackColor as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The colorcode of the background.
Notes: Color codes can be build up from red, blue, green and highlight.
red=1, blue=2, green=4 and highlight=8.
(Read and Write property)

34.1.4 CursorX as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The horizontal cursor position in characters.
Notes: (Read and Write property)

34.1.5   CursorY as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The vertical cursor position in characters.
Notes: (Read and Write property)

34.1.6   Height as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the window in characters.
Notes: (Read and Write property)

34.1.7   MaxHeight as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum height of the window in characters.
Notes: (Read and Write property)

34.1.8   MaxWidth as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The maximum width of the window in characters.
Notes: (Read and Write property)

34.1.9   TextColor as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The colorcode of the text.
Notes: Color codes can be build up from red, blue, green and highlight.
red=1, blue=2, green=4 and highlight=8.
(Read and Write property)

34.1.10   Width as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the window in characters.
Notes: (Read and Write property)

34.1.11   WindowHeight as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The height of the console window.
Notes: (Read and Write property)

34.1.12   WindowLeft as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The left position of the console window.
Notes: (Read and Write property)

34.1.13   WindowTop as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The top position of the console window.
Notes: (Read and Write property)

34.1.14   WindowWidth as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The width of the console window.
Notes: (Read and Write property)

34.2   class WindowsConsoleMBS

34.2.1   class WindowsConsoleMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for handling a console window in Windows.

34.2.2   Methods

34.2.3   Close

Plugin Version: 4.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources of this object now without waiting for Xojo to do it for you.

34.2.4   FlushConsole

Platform: Windows, Targets: Desktop, Console & Web.

Function: The FlushConsole function flushes the console input buffer.
Notes: All input records currently in the input buffer are discarded.

34.2.5   ReadConsole(maxcount as Integer) as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Reads maximum of maxcount characters from the input stream.

34.2.6   SetCursorPosition(x as Integer,y as Integer)

Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the cursor position inside the console window.

34.2.7 SetWindowPosition(absolute as boolean, left as Integer,top as Integer, right as Integer, bottom as Integer)

Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the window position and size.
Notes: If absolute=false, the coordinates are relative to the current one.

34.2.8 State as ConsoleStateMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: The current state of the console window.

34.2.9 WriteConsole(message as string) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes the message to the console.
Notes: Returns true if successfull.

34.2.10 Properties

34.2.11 AutoScrollAtEOL as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Automatical scrolling at end of line?
Notes: When writing with WriteConsole or echoing with ReadFile, the cursor moves to the beginning of the next row when it reaches the end of the current row. This causes the rows displayed in the console window to scroll up automatically when the cursor advances beyond the last row in the window. It also causes the contents of the screen buffer to scroll up (discarding the top row of the screen buffer) when the cursor advances beyond the last row in the screen buffer. If this mode is disabled, the last character in the row is overwritten with any subsequent characters.
(Read and Write property)

34.2.12 BackColor as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The text background color used.
Notes: (Read and Write property)

34.2.13   CursorSize as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The size of the cursor in percent (from 0 to 100).
Notes: (Read and Write property)

34.2.14   CursorVisible as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Is the cursor visible?
Notes: (Read and Write property)

34.2.15   EchoInput as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Echo the input on the screen?
Notes: Characters read by the ReadFile function are written to the active screen buffer as they are read.
This mode can be used only if the WaitForReturn mode is also enabled.
(Read and Write property)

34.2.16   GotConsole as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: whether the console was created inside the constructor.
Notes: (Read only property)

34.2.17   InputCodepage as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Codepage of the input stream.
Notes: (Read and Write property)

34.2.18   OutputCodepage as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Codepage of the output stream.
Notes: (Read and Write property)

34.2.19   ProcessInput as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Process input before reading?
Notes: Ctrl+c is processed by the system and is not placed in the input buffer. If the input buffer is being read by ReadConsole, other control keys are processed by the system and are not returned in the ReadConsole buffer. If the WaitForReturn mode is also enabled, backspace, carriage return, and linefeed characters are handled by the system.
(Read and Write property)

34.2.20   ProcessOutput as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Process output before reading?
Notes: Characters written by the WriteConsole function or echoed by the ReadFile or ReadConsole function are parsed for ASCII control sequences, and the correct action is performed. Backspace, tab, bell, carriage return, and linefeed characters are processed.
(Read and Write property)

34.2.21   TextColor as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The text color used.
Notes: (Read and Write property)

34.2.22   Title as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The title bar of the console window.
Notes: (Read and Write property)

34.2.23   WaitForReturn as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Wait for return key on reading?
Notes: The ReadConsole function returns only when a carriage return character is read. If this mode is disabled, the functions return when one or more characters are available.
(Read and Write property)

34.2.24   Events

34.2.25   ConsoleClosed

Platform: Windows, Targets: .

Function: Console was closed.

34.2.26   ConsoleOpened

Platform: Windows, Targets: .

Function: Console was opened.

34.2.27   ControlBreak as boolean

Platform: Windows, Targets: .

Function: Control-Break was pressed.
Notes: Return true if you handled it yourself.
Return false for the default handler.

34.2.28   ControlC as boolean

Platform: Windows, Targets: .

Function: Control-C was pressed.
Notes: Return true if you handled it yourself.
Return false for the default handler.

34.2.29   Logoff as boolean

Platform: Windows, Targets: .

Function: The user is logging off.
Notes: Return true if you handled it yourself.
Return false for the default handler.

34.2.30   Shutdown as boolean

Platform: Windows, Targets: .

Function: The user is shutting down the computer.
Notes: Return true if you handled it yourself.
Return false for the default handler.

34.2.31   UserClose as boolean

Platform: Windows, Targets: .

Function: The user pressed the close button.
Notes: Return true if you handled it yourself.
Return false for the default handler.

34.2.32   Constants

Constants

Constant	Value	Description
Black	0	The color code for Black. Read only property with value 0.
Blue	1	The color code for Blue. Read only property with value 2.
Green	2	The color code for Green. Read only property with value 4.
Highlight	8	The color code for Highlight. Read only property with value 8.
Red	4	The color code for Red. Read only property with value 1.

Chapter 35

Windows ICM

35.1   class WindowsICMColorMBS

35.1.1   class WindowsICMColorMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a color value.
Notes: A variable of type color may be accessed as any of the supported color space colors by accessing the appropriate member of the union.

see also:
http://msdn.microsoft.com/en-us/library/dd371932(v=VS.85).aspx
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.1.2   Properties

35.1.3   a as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The a color value of a Lab color.
Notes: (Read and Write property)

35.1.4   b as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The b color value of a Lab color.
Notes: (Read and Write property)

35.1.5   black as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The black color value of a CMYK color.
Notes: (Read and Write property)

35.1.6   blue as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The blue value.
Notes: Range from 0 to 255.
(Read and Write property)

35.1.7   ch1 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The first channel color value of a three channel color.
Notes: (Read and Write property)

35.1.8   ch2 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The second channel color value of a three channel color.
Notes: (Read and Write property)

35.1.9   ch3 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The third channel color value of a three channel color.
Notes: (Read and Write property)

35.1.10   cyan as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The cyan color value of a CMYK color.
Notes: (Read and Write property)

35.1.11   gray as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The gray color value.
Notes: For grayscale color space.
(Read and Write property)

35.1.12   green as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The green value.
Notes: Range from 0 to 255.
(Read and Write property)

35.1.13   Index as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The index value for an indexed color space.
Notes: (Read and Write property)

35.1.14   L as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The L color value of a Lab color.
Notes: (Read and Write property)

35.1.15   magenta as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The magenta color value of a CMYK color.
Notes: (Read and Write property)

35.1.16   red as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The red value.
Notes: Range from 0 to 255.
(Read and Write property)

35.1.17   XYZ_X as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The X value of a XYZ color.
Notes: (Read and Write property)

35.1.18   XYZ_Y as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y value of a XYZ color.
Notes: (Read and Write property)

35.1.19   XYZ_Z as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Z value of a XYZ color.
Notes: (Read and Write property)

35.1.20   yellow as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The yellow color value of a CMYK color.
Notes: (Read and Write property)

35.1.21   Yxy_x as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The x value of a Yxy color.
Notes: (Read and Write property)

35.1.22   Yxy_y as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The y value of a Yxy color.
Notes: Yxy_YY is the first Y and Yxy_Y the second one in a Yxy color.
(Read and Write property)

35.1.23   Yxy_YY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The y value of a Yxy color.
Notes: Yxy_YY is the first Y and Yxy_Y the second one in a Yxy color.
(Read and Write property)

35.1.24   Channel(index as Integer) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The channel value of a colorspaces with 3 to 8 channels.
Notes: Index from 0 to 7.
(Read and Write computed property)

35.1.25   Constants

Constants

Constant	Value	Description
MAX_COLOR_CHANNELS	8	The maximum number of supported color channels.

Color Type Constants

Constant	Value	Description
COLOR_3_CHANNEL	6	The COLOR is in the GENERIC3CHANNEL color space.
COLOR_5_CHANNEL	8	The COLOR is in a five channel color space.
COLOR_6_CHANNEL	9	The COLOR is in a six channel color space.
COLOR_7_CHANNEL	10	The COLOR is in a seven channel color space.
COLOR_8_CHANNEL	11	The COLOR is in an eight channel color space.
COLOR_CMYK	7	The COLOR is in the CMYKCOLOR color space.
COLOR_GRAY	1	The COLOR is in the GRAYCOLOR color space.
COLOR_Lab	5	The COLOR is in the LabCOLOR color space.
COLOR_NAMED	12	The COLOR is in a named color space.
COLOR_RGB	2	The COLOR is in the RGBCOLOR color space.
COLOR_XYZ	3	The COLOR is in the XYZCOLOR color space.
COLOR_Yxy	4	The COLOR is in the YxyCOLOR color space.

35.2 class WindowsICMEnumMBS

35.2.1 class WindowsICMEnumMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The WindowsICMEnumMBS class contains information that defines the profile enumeration constraints.
Example:

dim c as new WindowsICMEnumMBS // no options set

dim a(-1) as string = WindowsICMModuleMBS.EnumColorProfiles(c)

for each s as string in a
MsgBox s
next

Notes: see also
http://msdn.microsoft.com/en-us/library/dd316895(v=VS.85).aspx
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.2.2 Properties

35.2.3 Attributes0 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Attributes of profile that can be any of the following values.
Notes:

Constant	Meaning
ATTRIB_TRANSPARENCY	Turns transparency on. If this flag is not used, the attribute is reflective by default.
ATTRIB_MATTE	Turns matte display on. If this flag is not used, the attribute is glossy by default.

(Read and Write property)

35.2.4   Attributes1 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Attributes of profile that can be any of the following values.
Notes:

Constant	Meaning
ATTRIB_TRANSPARENCY	Turns transparency on. If this flag is not used, the attribute is reflective by default.
ATTRIB_MATTE	Turns matte display on. If this flag is not used, the attribute is glossy by default.

(Read and Write property)

35.2.5   Classs as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the profile class.
Notes: For a description of profile classes, see Using Device Profiles with WCS.
http://msdn.microsoft.com/en-us/library/dd372213(v=VS.85).aspx

A profile class may have any of the following values.

Profile Class	Signature
Input Device Profile	CLASS_SCANNER
Display Device Profile	CLASS_MONITOR
Output Device Profile	CLASS_PRINTER
Device Link Profile	CLASS_LINK
Color Space Conversion Profile	CLASS_COLORSPACE
Abstract Profile	CLASS_ABSTRACT
Named Color Profile	CLASS_NAMED
Color Appearance Model Profile	CLASS_CAMP
Color Gamut Map Model Profile	CLASS_GMMP

(Read and Write property)

35.2.6   CMMType as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identification number of the CMM that is used in the profile.
Notes: Identification numbers are registered with the ICC.
(Read and Write property)

35.2.7   ConnectionSpace as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A signature value that indicates the color space in which the profile connection space (PCS) is defined.
Notes: Can be any of the following values: SPACE_XYZ or SPACE_Lab

When the Classs member is set to CLASS_LINK, the PCS is taken from the DataColorSpace member.
(Read and Write property)

35.2.8   Creator as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Signature of the software that created the profile.
Notes: Signatures are registered with the ICC.
(Read and Write property)

35.2.9   DataColorSpace as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A signature value that indicates the color space in which the profile data is defined.
Notes: Can be any value from the SPACE_* Constants.
(Read and Write property)

35.2.10   DeviceClass as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the device class.
Notes: A device class may have one of the following values.

(Read and Write property)

Profile Class   Signature
 Input Device Profile   CLASS_SCANNER
 Display Device Profile   CLASS_MONITOR
 Output Device Profile   CLASS_PRINTER

35.2.11   DeviceName as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: User friendly name of the device.
Notes: (Read and Write property)

35.2.12   DitheringMode as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the style of dithering that will be used when an image is displayed.
Notes: (Read and Write property)

35.2.13   Fields as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates which fields in this class are being used.
Notes: Can be set to any combination of the following constant values:

ET_DEVICENAME
ET_MEDIATYPE
ET_DITHERMODE
ET_RESOLUTION
ET_CMMTYPE
ET_CLASS
ET_DATACOLORSPACE
ET_CONNECTIONSPACE
ET_SIGNATURE
ET_PLATFORM
ET_PROFILEFLAGS
ET_MANUFACTURER
ET_MODEL
ET_ATTRIBUTES
ET_RENDERINGINTENT
ET_CREATOR
ET_DEVICECLASS

(Read and Write property)

35.2.14 Manufacturer as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identification number of the device profile manufacturer.
Notes: All manufacturer identification numbers are registered with the ICC.
(Read and Write property)

35.2.15 MediaType as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates which type of media is associated with the profile, such as a printer or screen.
Notes: (Read and Write property)

35.2.16 Model as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device manufacturer's device model number.
Notes: All model identification numbers are registered with the ICC.
(Read and Write property)

35.2.17 Platform as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The primary platform for which the profile was created.
Notes: The member can be set to any of the following values.

Platform	Value
Apple Computer, Inc.	'APPL'
Microsoft Corp.	'MSFT'
Silicon Graphics, Inc.	'SGI'
Sun Microsystems, Inc.	'SUNW'
Taligent	'TGNT'

(Read and Write property)

35.2.18   ProfileFlags as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Bit flags containing hints that the CMM uses to interpret the profile data and can be set to one of the following values.
Notes:

Constant	Meaning
FLAG_EMBEDDEDPROFILE	The profile is embedded in a bitmap file.
FLAG_DEPENDENTONDATA	The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

(Read and Write property)

35.2.19   RenderingIntent as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The profile rendering intent.
Notes: The member can be set to one of the following values:

INTENT_PERCEPTUALINTENT_SATURATIONINTENT_RELATIVE_COLORIMETRICINTENT_AB-SOLUTE_COLORIMETRIC

For more information, see Rendering Intents.
http://msdn.microsoft.com/en-us/library/dd372183(v=VS.85).aspx
(Read and Write property)

35.2.20   ResolutionX as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The horizontal resolution in pixels of the device on which the image will be displayed.
Notes: (Read and Write property)

35.2.21 ResolutionY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The vertical resolution in pixels of the device on which the image will be displayed.
Notes: (Read and Write property)

35.2.22 Signature as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reserved for internal use.
Notes: (Read and Write property)

35.2.23 Constants

Attribute Constants

Constant	Value	Description
ATTRIB_MATTE	2	Turns matte display on. If this flag is not used, the attribute is glossy by default.
ATTRIB_TRANSPARENCY	1	Turns transparency on. If this flag is not used, the attribute is reflective by default.

Profile Class Constants

Constant	Value	Description
CLASS_ABSTRACT	&h61627374	
CLASS_CAMP	&h6C616D70	
CLASS_COLORSPACE	&h73706163	
CLASS_GMMP	&h676D6D70	
CLASS_LINK	&h6C696E6B	
CLASS_MONITOR	&h6D6E7472	
CLASS_NAMED	&h6E6D636C	
CLASS_PRINTER	&h70727472	
CLASS_SCANNER	&h73636E72	

Field usage flags

Constant	Value	Description
ET_ATTRIBUTES	&h02000	
ET_CLASS	&h00020	
ET_CMMTYPE	&h00010	
ET_CONNECTIONSPACE	&h00080	
ET_CREATOR	&h08000	
ET_DATACOLORSPACE	&h00040	
ET_DEVICECLASS	&h10000	
ET_DEVICENAME	&h00001	
ET_DITHERMODE	&h00004	
ET_MANUFACTURER	&h00800	
ET_MEDIATYPE	&h00002	
ET_MODEL	&h01000	
ET_PLATFORM	&h00200	
ET_PROFILEFLAGS	&h00400	
ET_RENDERINGINTENT	&h04000	
ET_RESOLUTION	&h00008	
ET_SIGNATURE	&h00100	

Profile flag constants.

Constant	Value	Description
FLAG_DEPENDENTONDATA	2	The profile can't be used independently of the embed profiles that are embedded in bitmap files.
FLAG_EMBEDDEDPROFILE	1	The profile is embedded in a bitmap file.
FLAG_ENABLE_CHROMATIC_ADAPTATION	&h02000000	

Platform signature

Constant	Value	Description
SigMacintosh	&h4150504C	APPL = Apple Computer, Inc.
SigMicrosoft	&h4D534654	MSFT = Microsoft Corp.
SigSGI	&h53474920	SGI = Silicon Graphics, Inc.
SigSolaris	&h53554E57	SUNW = Sun Microsystems, Inc.
SigTaligent	&h54474E54	TGNT = Taligent

Color Space Constants

Constant	Value	Description
SPACE_Lab	&h4C616220	Lab
SPACE_XYZ	&h58595A20	XYZ

35.3 class WindowsICMLogColorSpaceMBS

35.3.1 class WindowsICMLogColorSpaceMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class contains information that defines a logical color space.
Notes: If the CSType member is set to LCS_sRGB or LCS_WINDOWS_COLOR_SPACE, the other members of this structure are ignored, and WCS uses the sRGB color space. The Endpoints, GammaRed, GammaGreen, and GammaBlue members are used to describe the logical color space. The Endpoints member is a CIEXYZTRIPLE that contains the x, y, and z values of the color space's RGB endpoint.

The required DWORD bit format for the GammaRed, GammaGreen, and GammaBlue is an 8.8 fixed point interger left-shifted by 8 bits. The plugin takes care about that detail.

Whenever the Filename member contains a file name and the CSType member is set to LCS_CALIBRATED_RGB, WCS ignores the other members of this class. It uses the color space in the file as the color space to which this LOGCOLORSPACE structure refers.

The relation between tri-stimulus values X,Y,Z and chromaticity values x,y,z is as follows:

x = X/(X+Y+Z)
y = Y/(X+Y+Z)
z = Z/(X+Y+Z)

If the CSType member is set to LCS_sRGB or LCS_WINDOWS_COLOR_SPACE, the other members of this structure are ignored, and ICM uses the sRGB color space. Appliations should still initialize the rest of the structure since CreateProfileFromLogColorSpace ignores CSType member and uses Endpoints, GammaRed, GammaGreen, GammaBlue members to create a profile, which may not be initialized in case of LCS_sRGB or LCS_WINDOWS_COLOR_SPACE color spaces.
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.3.2 Properties

35.3.3 CSType as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Color space type.
Notes: The member can be one of the following values.

Value	Meaning
LCS_CALIBRATED_RGB	Color values are calibrated RGB values. The values are translated using the endpoints specified by the lcsEndpoints member before being passed to the device.
LCS_sRGB	Color values are values are sRGB values.
LCS_WINDOWS_COLOR_SPACE	Color values are Windows default color space color values.

(Read and Write property)

35.3.4   EndpointsBX as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The X value of the Blue Endpoint.
Notes: (Read and Write property)

35.3.5   EndpointsBY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y value of the Blue Endpoint.
Notes: (Read and Write property)

35.3.6   EndpointsBZ as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Z value of the Blue Endpoint.
Notes: (Read and Write property)

35.3.7   EndpointsGX as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The X value of the Green Endpoint.
Notes: (Read and Write property)

35.3.8   EndpointsGY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y value of the Green Endpoint.
Notes: (Read and Write property)

35.3.9   EndpointsGZ as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Z value of the Green Endpoint.
Notes: (Read and Write property)

35.3.10   EndpointsRX as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The X value of the Red Endpoint.
Notes: (Read and Write property)

35.3.11   EndpointsRY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Y value of the Red Endpoint.
Notes: (Read and Write property)

35.3.12   EndpointsRZ as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Z value of the Red Endpoint.
Notes: (Read and Write property)

35.3.13   Filename as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string that names a color profile file.
Notes: This member is typically set to "", but may be used to set the color space to be exactly as specified by the color profile. This is useful for devices that input color values for a specific printer, or when using an installable image color matcher. If a color profile is specified, all other members of this class should be set to reasonable values, even if the values are not completely accurate.
(Read and Write property)

35.3.14   GammaBlue as Double

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scale of the blue coordinate.
Notes: (Read and Write property)

35.3.15   GammaGreen as Double

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scale of the green coordinate.
Notes: (Read and Write property)

35.3.16   GammaRed as Double

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Scale of the red coordinate.
Notes: (Read and Write property)

35.3.17   Intent as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The gamut mapping method.
Notes: This member can be one of the following values.

Value	Value	Intent	ICC Name	Meaning
LCS_GM_BUSINESS	1	Graphic	Saturation	Maintain saturation. Used for business charts and other situations in which undithered colors are required.
LCS_GM_GRAPHICS	2	Proof	Relative Colorimetric	Maintain colorimetric match. Used for graphic designs and named colors.
LCS_GM_IMAGES	4	Picture	Perceptual	Maintain contrast. Used for photographs and natural images.
LCS_GM_ABS_COLORIMETRIC	8	Match	Absolute Colorimetric	Maintain the white point. Match the colors to their nearest color in the destination gamut.

(Read and Write property)

35.3.18   Constants

Rendering Intent Constants

Constant	Value	Description
INTENT_ABSOLUTE_COLORIMETRIC	3	Maintain the white point. Match the colors to their nearest color i nation gamut.
INTENT_PERCEPTUAL	0	Maintain contrast. Used for photographs and natural images.
INTENT_RELATIVE_COLORIMETRIC	1	Maintain colorimetric match. Used for graphic designs and named
INTENT_SATURATION	2	Maintain saturation. Used for business charts and other situation undithered colors are required.

Color Space Type Constants

Constant	Value	Description
LCS_CALIBRATED_RGB	0	Color values are calibrated RGB values. The values are translate endpoints specified by the Endpoints member before being passed to
LCS_sRGB	&h73524742	Color values are values are sRGB values.
LCS_WINDOWS_COLOR_SPACE	&h57696E20	Color values are Windows default color space color values.

35.4 module WindowsICMModuleMBS

35.4.1 module WindowsICMModuleMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The main module for Windows Image Color Matching functions.
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.4.2 Methods

35.4.3 AssociateColorProfileWithDevice(ProfileName as string, DeviceName as string) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The AssociateColorProfileWithDevice function associates a specified color profile with a specified device.
Notes: ProfileName: The file name of the profile to associate.
DeviceName: The name of the device to associate.

Returns true on success and false on failure.

The AssociateColorProfileWithDevice function will fail if the profile has not been installed on the computer using the InstallColorProfile function.

Note that under Windows (Windows 95 or later), the PostScript device driver for printers assumes a CMYK color model. Therefore, all PostScript printers must use a CMYK color profile. Windows 2000 does not have this limitation.

If the specified device is a monitor, this function updates the default profile.

Several profiles are typically associated with printers, based on paper and ink types. There is no default. The GDI selects the best one from the associated profiles when your application creates a device context (DC).

Scanners also have no default profile. However, it is atypical to associate more than one profile with a scanner.

AssociateColorProfileWithDevice always adds the specified profile to the current user's per-user profile association list for the specified device. Before adding the profile to the list, AssociateColorProfileWithDevice determines whether the user has previously expressed the desire to use a per-user profile association list

for the device. If so, then AssociateColorProfileWithDevice simply adds the specified profile to the existing per-user profile association list for the device. If not, then AssociateColorProfileWithDevice creates a new per-user profile association list for the device by copying the system-wide association list for that device. It then appends the specified profile to the per-user list. From that point on, the current user will be using a per-user profile association list for the specified device, as if SetUsePerUserProfiles had been called for Device with the usePerUserProfiles parameter set to TRUE.

35.4.4 DisassociateColorProfileFromDevice(ProfileName as string, DeviceName as string) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The DisassociateColorProfileFromDevice function disassociates a specified color profile with a specified device on a specified computer.
Notes: Returns true on success and false on failure.

ProfileName: The file name of the profile to disassociate.
DeviceName: The name of the device to disassociate.

If more than one profile is associated with a device, WCS uses the last one associated as the default. That is, if your application sequentially associates three profiles with a device, WCS will use the last one associated as the default. If your application then calls the DisassociateColorProfileFromDevice function to disassociate the third profile (which is the default in this example), the WCS will use the second profile as the default.

If your application disassociates all profiles from a device, WCS uses the sRGB profile as the default.

DisassociateColorProfileFromDevice always removes the specified profile from the current user's per-user profile association list for the specified device. Before removing the profile from the list, DisassociateColorProfileFromDevice determines whether the user has previously expressed the desire to use a per-user profile association list for the device. If so, then DisassociateColorProfileFromDevice simply removes the specified profile from the existing per-user profile association list for the device. If not, then DisassociateColorProfileFromDevice creates a new per-user profile association list for the device by copying the system-wide association list for that device. It then removes the specified profile from the per-user list. From that point on, the current user will be using a per-user profile association list for the specified device, as if SetUsePerUserProfiles had been called for Device with the usePerUserProfiles parameter set to TRUE.

35.4.5 EnumColorProfiles(criterias as WindowsICMEnumMBS) as string()

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The EnumColorProfiles function enumerates all the profiles satisfying the given enumeration criteria.

Example:

dim c as new WindowsICMEnumMBS // no options set

dim a(-1) as string = WindowsICMModuleMBS.EnumColorProfiles(c)

for each s as string in a
MsgBox s
next

Notes: On success the function returns an array of profile names.

Several profiles are typically associated with printers, based on the paper and ink types. There is a default profile for each device. For International Color Consortium (ICC) profiles, GDI selects the best one from the ICC-associated profiles when your application creates a device context (DC).

Do not attempt to use EnumColorProfiles to determine the default profile for a device. Instead, create a device context for the device and then invoke the GetICMProfile function. On Windows Vista and Windows 7, the WcsGetDefaultColorProfile function can also be used to determine a device's default color profile.

If the Fields member of WindowsICMEnumMBS that is pointed to by the criterias parameter is set to ET_DEVICENAME, this function will enumerate all of the color profiles associated with all types of devices attached to the user's computer, regardless of the device class. If the Fields member is set to ET_DEVICENAME or ET_DEVICECLASS and a device class is specified in the DeviceClass member, this function will only enumerate the profiles associated with the specified device class. If the Fields member is set only to ET_DEVICECLASS, the EnumColorProfiles function will enumerate all profiles that can be associated with that type of device.

Whenever EnumColorProfiles is examining the profiles associated with a specific device, the results depend on whether the user has chosen to use the system-wide list of profiles associated with that device, or his or her own ("per-user") list. Calling SetUsePerUserProfiles with its usePerUserProfiles parameter set to TRUE causes future calls to EnumColorProfiles to look at only the current user's per-user list of profile associations for the specified device. Calling WcsSetUsePerUserProfiles with its usePerUserProfiles parameter set to FALSE causes future calls to EnumColorProfiles to look at the system-wide list of profile associations for the specified device. If SetUsePerUserProfiles has never been called for the current user, EnumColorProfiles examines the system-wide list.

This function will provide the information for converting WCS-specific DMP information to the legacy EnumType record in enable consistent profile enumeration. The defaults will be the same as ICC if this information is not present.

Per-user/LUA support

The enumeration is specific to current user. Both system wide and current user device associations are considered. For default profile configuration, current user settings override system wide ones.

35.4.6 GetColorDirectory as folderitem

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetColorDirectory function retrieves the path of the Windows COLOR directory on a specified machine.
Example:

dim f as FolderItem = WindowsICMModuleMBS.GetColorDirectory

if f=nil then
MsgBox "No path?"
else
MsgBox f.NativePath
end if

Notes: On success returns folderitem for color directory. Returns nil on any error.

35.4.7 GetStandardColorSpaceProfile(ProfileID as Integer) as string

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetStandardColorSpaceProfile function retrieves the color profile registered for the specified standard color space.
Example:

MsgBox WindowsICMModuleMBS.GetStandardColorSpaceProfile(WindowsICMModuleMBS.LCS_WINDOWS_COLOR_S

Notes: ProfileID: Specifies the ID value of the standard color space for which to retrieve the profile. The only valid values for this parameter are LCS_sRGB and LCS_WINDOWS_COLOR_SPACE.

This function supports Windows Color System (WCS) device model profiles (DMPs) in addition to International Color Consortium (ICC) profiles. It does not support WCS CAMP or GMMP profiles and will return an error if such profiles are used.

Overview of Windows Vista Specific Functionality

This will support WCS DMPs in addition to ICC profiles. It will not support WCS CAMP or GMMP profiles and will return an error if such profiles are used with this API.

Per-user/LUA support

This will retrieve the color profile registered for the given standard color space for current user. If there is no such setting for the current user, it retrieves the system wide setting.

This uses WcsGetDefaultColorProfile with WCS_PROFILE_MANAGEMENT_SCOPE_CURRENT_USER.

This is executable in LUA context.

35.4.8   InstallColorProfile(file as folderitem) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The InstallColorProfile function installs a given profile for use on a specified machine.
Notes: The profile is also copied to the COLOR directory.
Returns true on success and false on failure.

35.4.9   RegisterCMM(cmmID as Integer, file as folderitem) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: RegisterCMM associates a specified identification value with the specified color management module dynamic link library (CMM DLL).
Notes: When this ID appears in a color profile, Windows can then locate the corresponding CMM so as to create a transform.

cmmID: Specifies the ID signature of the CMM registered with the International Color Consortium (ICC).

file: Points to the CMM DLL.

Returns true on success and false on failure.

35.4.10   SelectCMM(cmmID as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: SelectCMM allows an application to select the preferred color management module (CMM) to use.
Notes: cmmID: Specifies the signature of the desired CMM as registered with the International Color Consortium (ICC).

Windows 2000 only: Setting this parameter to 0 causes the WCS system to select the default CMM.

Returns true on success and false on failure.

35.4.11   SetStandardColorSpaceProfile(ProfileID as Integer, ProfileName as folderitem) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetStandardColorSpaceProfile function registers a specified profile for a given standard color space.
Notes: The profile can be queried using GetStandardColorSpaceProfile.

ProfileID: Specifies the ID value of the standard color space that the given profile represents.
Profilename: path to the profile file.

Returns true on success and false on failure.

The profile must already be installed on the system before it can be registered for a standard color space.

This function supports Windows Color System (WCS) device model profiles (DMPs) in addition to International Color Consortium (ICC) profiles. It does not support WCS CAMP or GMMP profiles and will return an error if such profiles are used.

Per-user/LUA support

This will register a specified profile for a given standard color space only for current user.

This uses SetDefaultColorProfile with WCS_PROFILE_MANAGEMENT_SCOPE_CURRENT_USER.

This is executable in LUA context if the profile is already installed, fails otherwise with access denied since install is system-wide and requires administrator privileges.

35.4.12 UninstallColorProfile(ProfileName as string, DeleteFile as boolean = true) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: UninstallColorProfile removes a specified color profile from a specified computer. Associated files are optionally deleted from the system.
Notes: ProfileName: Points to the file name of the profile to uninstall.

DeleteFile: If set to true, the function deletes the profile from the COLOR directory. If set to false, this function has no effect.

Returns true on success and false on failure.

35.4.13 UnregisterCMM(cmmID as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The UnregisterCMM function dissociates a specified ID value from a given color management module dynamic-link library (CMM DLL).
Notes: cmmID: Specifies the ID value identifying the CMM whose registration is to be removed. This is the signature of the CMM registered with the International Color Consortium (ICC).

Returns true on success and false on failure.

35.4.14 Constants

Info selectors for GetInfo

Constant	Value	Description
CMM_DESCRIPTION	5	A text string that describes the color management module.
CMM_DLL_VERSION	3	Version number of the CMM DLL.
CMM_DRIVER_VERSION	2	
CMM_IDENT	1	The CMM identification signature registered with the International Color Consortium (ICC).
CMM_LOGOICON	6	The logo icon for this CMM.
CMM_VERSION	4	Version of Windows supported.
CMM_WIN_VERSION	0	Backward compatibility with Windows 95.

Profile ID Constants

Constant	Value	Description
LCS_sRGB	&h73524742	
LCS_WINDOWS_COLOR_SPACE	&h57696E20	

35.5   class WindowsICMNamedProfileInfoMBS

35.5.1   class WindowsICMNamedProfileInfoMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class is used to store information about a named color profile.
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.5.2   Properties

35.5.3   Count as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Total number of named colors in the profile.
Notes: (Read and Write property)

35.5.4   CountDevCoordinates as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Total number of device coordinates for each named color.
Notes: (Read and Write property)

35.5.5   Flags as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Flags for this information record.
Notes: Not currently used by the default CMM.
(Read and Write property)

35.5.6   Prefix as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string containing the prefix for each color name.
Notes: (Read and Write property)

35.5.7 Suffix as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string containing the suffix for each color name.
Notes: (Read and Write property)

35.6 class WindowsICMProfileHeaderMBS

35.6.1 class WindowsICMProfileHeaderMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class contains information that describes the contents of a device profile file.
Notes: This header occurs at the beginning of a device profile file.
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.6.2 Properties

35.6.3 Attributes0 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Attributes of profile.
Notes: (Read and Write property)

35.6.4 Attributes1 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Attributes of profile.
Notes: (Read and Write property)

35.6.5 Classs as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Indicates the profile class.
Example:

```
// some profile file
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")

// open profile read only
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-
ING)

// get headers
```

```
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

// show color space name
Select case h.Classs
case WindowsICMProfileHeaderMBS.CLASS_MONITOR
msgbox "Monitor"
case WindowsICMProfileHeaderMBS.CLASS_PRINTER
msgbox "Printer"
case WindowsICMProfileHeaderMBS.CLASS_SCANNER
msgbox "Scanner"
case WindowsICMProfileHeaderMBS.CLASS_LINK
msgbox "Link"
case WindowsICMProfileHeaderMBS.CLASS_ABSTRACT
msgbox "Abstract"
case WindowsICMProfileHeaderMBS.CLASS_COLORSPACE
msgbox "Colorspace"
case WindowsICMProfileHeaderMBS.CLASS_NAMED
msgbox "Named"
case WindowsICMProfileHeaderMBS.CLASS_CAMP
msgbox "Camp"
case WindowsICMProfileHeaderMBS.CLASS_GMMP
msgbox "GNMP"
else
msgbox "Unknown: "+hex(h.Classs)
end Select
```

Notes: For a description of profile classes, see Using Device Profiles with WCS: http://msdn.microsoft.com/en-us/library/dd372213(v=VS.85).aspx

A profile class may have any of the values from the CLASS_* constants.

Class is written with three s here as Class is a reserved word in Xojo.
(Read and Write property)

35.6.6 CMMType as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identification number of the CMM that is used in the profile.
Notes: Identification numbers are registered with the ICC.
(Read and Write property)

35.6.7   ConnectionSpace as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A signature value that indicates the color space in which the profile connection space (PCS) is defined.
Notes: The member can be any of the following values: SPACE_XYZ or SPACE_Lab.
(Read and Write property)

35.6.8   Creator as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Signature of the software that created the profile.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-
ING)
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

MsgBox hex(h.Creator)

Notes: Signatures are registered with the ICC.
(Read and Write property)

35.6.9   DataColorSpace as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A signature value that indicates the color space in which the profile data is defined.
Example:

function Name() as string
dim w as WindowsICMProfileMBS // your profile
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

if h=nil then Return "?"

Select case h.DataColorSpace
case WindowsICMProfileHeaderMBS.SPACE_XYZ
Return "XYZ"
case WindowsICMProfileHeaderMBS.SPACE_Lab

Return "Lab"
case WindowsICMProfileHeaderMBS.SPACE_Luv
Return "Luv"
case WindowsICMProfileHeaderMBS.SPACE_YCbCr
Return "YCbCr"
case WindowsICMProfileHeaderMBS.SPACE_Yxy
Return "Yxy"
case WindowsICMProfileHeaderMBS.SPACE_RGB
Return "RGB"
case WindowsICMProfileHeaderMBS.SPACE_GRAY
Return "GRAY"
case WindowsICMProfileHeaderMBS.SPACE_HSV
Return "HSV"
case WindowsICMProfileHeaderMBS.SPACE_HLS
Return "HLS"
case WindowsICMProfileHeaderMBS.SPACE_CMYK
Return "CMYK"
case WindowsICMProfileHeaderMBS.SPACE_CMY
Return "CMY"
case WindowsICMProfileHeaderMBS.SPACE_2_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_3_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_4_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_5_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_6_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_7_CHANNEL
Return "2 Channel"
case WindowsICMProfileHeaderMBS.SPACE_8_CHANNEL
Return "2 Channel"
else
Return "Unknown: "+hex(h.DataColorSpace)
end Select
end function

Notes: The member can be any of value from the SPACE_* Constants.
(Read and Write property)

35.6.10 DateTime0 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data and time that the profile was created.
Notes: (Read and Write property)

35.6.11   DateTime1 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data and time that the profile was created.
Notes: (Read and Write property)

35.6.12   DateTime2 as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The data and time that the profile was created.
Notes: (Read and Write property)

35.6.13   IlluminantX as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: X value of the Profile illuminant.
Notes: (Read and Write property)

35.6.14   IlluminantY as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Y value of the Profile illuminant.
Notes: (Read and Write property)

35.6.15   IlluminantZ as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Z value of the Profile illuminant.
Notes: (Read and Write property)

35.6.16 Manufacturer as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The identification number of the device profile manufacturer.
Notes: All manufacturer identification numbers are registered with the ICC.
(Read and Write property)

35.6.17 Model as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The device manufacturer's device model number.
Notes: All model identification numbers are registered with the ICC.
(Read and Write property)

35.6.18 Platform as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The primary platform for which the profile was created.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-
ING)
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

MsgBox DecodingFromHexMBS(hex(h.Platform)) // shows platform, e.g. "APPL"

Notes: The primary platform can be set to any of the following values.

Platform	Value
Apple Computer, Inc.	'APPL'
Microsoft Corp.	'MSFT'
Silicon Graphics, Inc.	'SGI'
Sun Microsystems, Inc.	'SUNW'
Taligent	'TGNT'

(Read and Write property)

35.6.19 ProfileFlags as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Bit flags containing hints that the CMM uses to interpret the profile data.
Notes: The member can be set to the following values.

Constant	Meaning
FLAG_EMBEDDEDPROFILE	The profile is embedded in a bitmap file.
FLAG_DEPENDENTONDATA	The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

(Read and Write property)

35.6.20 RenderingIntent as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The profile rendering intent.
Notes: The member can be set to one of the following values:

INTENT_PERCEPTUALINTENT_SATURATIONINTENT_RELATIVE_COLORIMETRICINTENT_AB-SOLUTE_COLORIMETRIC

For more information, see Rendering Intents.
http://msdn.microsoft.com/en-us/library/dd372183(v=VS.85).aspx
(Read and Write property)

35.6.21 Signature as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Reserved for internal use.
Notes: (Read and Write property)

35.6.22 Version as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The version number of the profile.
Notes: The version number is determined by the ICC. The current major version number is &h02. The

current minor version number is &h10. The major and minor version numbers are in binary coded decimal (BCD). They must be stored in the following format.

Byte Number	Contents
0	Major version number in BCD.
1	Minor version number in the most significant nibble of this byte. Bug fix version number in the least significant nibble.
2	Reserved. Must be set to 0.
3	Reserved. Must be set to 0.

(Read and Write property)

35.6.23   Constants

Attribute Constants

Constant	Value	Description
ATTRIB_MATTE	2	Turns matte display on. If this flag is not used, the attribute is glossy by default.
ATTRIB_TRANSPARENCY	1	Turns transparency on. If this flag is not used, the attribute is reflective by default.

Profile Class Constants

Constant	Value	Description
CLASS_ABSTRACT	&h61627374	Abstract Profile
CLASS_CAMP	&h6C616D70	Color Appearance Model Profile
CLASS_COLORSPACE	&h73706163	Color Space Conversion Profile
CLASS_GMMP	&h676D6D70	Color Gamut Map Model Profile
CLASS_LINK	&h6C696E6B	Device Link Profile
CLASS_MONITOR	&h6D6E7472	Display Device Profile
CLASS_NAMED	&h6E6D636C	Named Color Profile
CLASS_PRINTER	&h70727472	Output Device Profile
CLASS_SCANNER	&h73636E72	Input Device Profile

Profile Flags

Constant	Value	Description
FLAG_DEPENDENTONDATA	2	The profile can't be used independently of the embedded profiles that are embedded in bitmap files.
FLAG_EMBEDDEDPROFILE	1	The profile is embedded in a bitmap file.
FLAG_ENABLE_CHROMATIC_ADAPTATION	&h02000000	

Color Space Constants

Constant	Value	Description
SPACE_2_CHANNEL	&h32434C52	Generic 2 channel
SPACE_3_CHANNEL	&h33434C52	Generic 3 channel
SPACE_4_CHANNEL	&h34434C52	Generic 4 channel
SPACE_5_CHANNEL	&h35434C52	Generic 5 channel
SPACE_6_CHANNEL	&h36434C52	Generic 6 channel
SPACE_7_CHANNEL	&h37434C52	Generic 7 channel
SPACE_8_CHANNEL	&h38434C52	Generic 8 channel
SPACE_CMY	&h434D5920	CMY
SPACE_CMYK	&h434D594B	CMYK
SPACE_GRAY	&h47524159	Gray scale
SPACE_HLS	&h484C5320	HLS
SPACE_HSV	&h48535620	HSV
SPACE_Lab	&h4C616220	Lab
SPACE_Luv	&h4C757620	Luv
SPACE_RGB	&h52474220	RGB
SPACE_XYZ	&h58595A20	XYZ
SPACE_YCbCr	&h59436272	YCbCr
SPACE_Yxy	&h59787920	Yxy

35.7 class WindowsICMProfileMBS

35.7.1 class WindowsICMProfileMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a color profile.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.1pr4

- MBS Xojo / Real Studio Plugins, version 14.1pr3

- MBS Real Studio Plugins, version 11.1pr11

35.7.2 Methods

35.7.3 ConvertColorNameToIndex(name as string) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CMConvertColorNameToIndex function converts color names in a named color space to index numbers in a color profile.
Notes: name: The name of the color.

Returns the color index.

This function is required in the default CMM. It is optional for all other CMMs.

35.7.4 ConvertIndexToColorName(index as Integer) as string

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The ConvertIndexToColorName transforms indices in a color space to an array of names in a named color space.
Notes: This function is required in the default CMM. It is optional for all other CMMs.

35.7.5 CountColorProfileElements as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetCountColorProfileElements function retrieves the number of tagged elements in a given color profile.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-ING)

MsgBox "CountColorProfileElements: "+str(w.CountColorProfileElements)

Notes: Returns number of tagged elements in the profile or 0 on any error.
This function will fail if hProfile is not a valid ICC profile.
This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP.

35.7.6   CreateIccProfile(options as Integer = 0) as WindowsICMProfileMBS

Plugin Version: 14.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Converts a WCS profile into an International Color Consortium (ICC) profile.
Notes: Options: A flag value that specifies the profile conversion options.By default, the original WCS profiles used for the conversion are embedded in the output ICC profile in a Microsoft private tag, ProfilesTag (with signature "MS000". This produces an ICC profile that is compatible with ICC software, yet retains the original WCS profile data available to code designed to parse it.The possible values of this parameter are as follows. Any bits not defined in this list are reserved and should be set to zero:

WCS_DEFAULT	Specifies that the new ICC profile contains the original WCS profile in a private ProfilesTag.
WCS_ICCONLY	Specifies that the new ICC profile does not contain either the ProfilesTag or the original WCS profile.

Returns new profile object.

see also
http://msdn.microsoft.com/en-us/library/windows/desktop/dd372239(v=vs.85).aspx

35.7.7   GetColorProfileElement(tag as Integer) as string

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetColorProfileElement function copies data from a specified tagged profile element of a

specified color profile into a buffer.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("sRGB Profile.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-ING)

MsgBox w.GetColorProfileElement(&h64657363) // that's the code for desc, the description

Notes: tag: Identifies the tagged element from which to copy.

This function will fail if Profile is not a valid International Color Consortium (ICC) profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with, and hard coded to, ICC tag types and there exist many robust XML parsing libraries.

35.7.8 GetColorProfileElementTag(index as Integer) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetColorProfileElementTag function retrieves the tag name specified by dwIndex in the tag table of a given International Color Consortium (ICC) color profile, where Index is a one-based index into that table.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-ING)
dim list(-1) as string
dim c as Integer = w.CountColorProfileElements-1

for i as Integer = 0 to c
list.Append DecodingFromHexMBS(hex(w.GetColorProfileElementTag(i)))
next

MsgBox "Tags: "+Join(list, ", ")

Notes: Index: Specifies the one-based index of the tag to retrieve.

This function will fail if Profile is not a valid ICC profile.

GetColorProfileElementTag can be used to enumerate all tags in a profile after getting the number of tags in the profile using GetCountColorProfileElements.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with, and hard coded to, ICC tag types and there exist many robust XML parsing libraries.

35.7.9   GetNamedProfileInfo as WindowsICMNamedProfileInfoMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetNamedProfileInfo function retrieves information about the International Color Consortium (ICC) named color profile that is specified in the first parameter.
Notes: This function will fail if hProfile is not a valid ICC profile.
This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because named profiles are explicit ICC profile types.
Returns nil on any error.

35.7.10   GetProfileData as string

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Given valid color profile, the GetProfileData function will return the contents of the profile.
Notes: If the object is a Windows Color System (WCS) handle, then the DMP is returned and the CAMP and GMMP associated with the HPROFILE are ignored.

Returns the data or an empty string on any error.
See GetColorProfileFromHandle in MSDN documentation.

35.7.11   IsColorProfileTagPresent(tag as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IsColorProfileTagPresent function reports whether a specified International Color Consortium (ICC) tag is present in the specified color profile.
Notes: tag: Specifies the ICC tag to check.

Returns true if the tag is valid and false if not.

This function will fail if Profile is not a valid ICC profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with and hard coded to ICC tag types and there exist many robust XML parsing libraries.

35.7.12 IsValid as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The IsValid function reports whether the given profile is a valid ICC profile that can be used for color management.
Example:

dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-ING)

MsgBox "Valid: "+str(w.IsValid)

Notes: Returns true if the profile is valid.

Only the Windows default CMM is required to export this function; it is optional for all other CMMs.

If a CMM does not support this function, Windows uses the default CMM to validate the profile.

35.7.13 OpenProfileData(data as string, DesiredAccess as Integer) as WindowsICM-ProfileMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The OpenColorProfile function opens or create a color profile.
Notes: file: The file where to load profile from.
DesiredAccess: Specifies how to access the given profile. This parameter must take one the following constant values.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default

Value	Meaning
PROFILE_READ	Opens the profile for read access.
PROFILE_READWRITE	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

CAMP and GMMP in the registry.

When OpenColorProfile encounters an ICC profile with an embedded WCS profile, and if the dwType member within the Profile structure does not take the value DONT_USE_EMBEDDED_WCS_PROFILES, it should extract and use the WCS profile(s) contained in this WcsProfilesTag. The HPROFILE returned would be a WCS HPROFILE.

When the function opens the ICC profile, it will look for a WcsProfilesTag and, if there is one, it will extract and use the original WCS profiles contained therein. (See WcsCreateIccProfile.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default GMMP from the registry. An HPROFILE is a composition of a DMP, CAMP and GMMP.

35.7.14   OpenProfileFile(file as folderitem, DesiredAccess as Integer, ShareMode as Integer, CreationMode as Integer) as WindowsICMProfileMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The OpenColorProfile function opens or create a color profile.
Notes: file: The file where to load profile from.
DesiredAccess: Specifies how to access the given profile. This parameter must take one the following constant values.

Value	Meaning
PROFILE_READ	Opens the profile for read access.
PROFILE_READWRITE	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

ShareMode: Specifies how the profile should be shared, if the profile is contained in a file. A value of zero prevents the profile from being shared at all. The parameter can contain one or both of the following constants (combined by addition or logical OR).

CreationMode: Specifies which actions to take on the profile while opening it, if it is contained in a file. This parameter must take one of the following constant values.

Value	Meaning
FILE_SHARE_READ	Other open operations can be performed on the profile for read access.
FILE_SHARE_WRITE	Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

Value	Meaning
CREATE_NEW	Creates a new profile. Fails if the profile already exists.
CREATE_ALWAYS	Creates a new profile. Overwrites the profile if it exists.
OPEN_EXISTING	Opens the profile. Fails if it does not exist
OPEN_ALWAYS	Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.
TRUNCATE_EXISTING	Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default CAMP and GMMP in the registry.

When OpenColorProfile encounters an ICC profile with an embedded WCS profile, and if the dwType member within the Profile structure does not take the value DONT_USE_EMBEDDED_WCS_PROFILES, it should extract and use the WCS profile(s) contained in this WcsProfilesTag. The HPROFILE returned would be a WCS HPROFILE.

CreationMode flags CREATE_NEW, CREATE_ALWAYS, and TRUNCATE_EXISTING, will always return blank ICC HPROFILEs. If other CreationMode flags are present, InternalOpenColorProfile is called (using the flags as provided by the API) to determine whether the profile is ICC or WCS XML.

Within the ICC code path, an ICC Profile is returned using the requested sharing, access and creation flags as specified in the tables above.

Within the WCS path, the CreationMode flag OPEN_ALWAYS will fail if the profile doesn't exist, since WCS profiles cannot be created or edited within the WCS architecture (they must be edited outside of it, using MSXML6). For the same reason, dwShareMode flag FILE_SHARE_WRITE, and dwDesiredAccess flag PROFILE_READWRITE are ignored within the WCS path.

When the function opens the ICC profile, it will look for a WcsProfilesTag and, if there is one, it will extract and use the original WCS profiles contained therein. (See WcsCreateIccProfile.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default

GMMP from the registry. An HPROFILE is a composition of a DMP, CAMP and GMMP.

35.7.15   OpenProfilePath(path as string, DesiredAccess as Integer, ShareMode as Integer, CreationMode as Integer) as WindowsICMProfileMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The OpenColorProfile function opens or create a color profile.
Notes: file: The file where to load profile from.
DesiredAccess: Specifies how to access the given profile. This parameter must take one the following constant values.

Value	Meaning
PROFILE_READ	Opens the profile for read access.
PROFILE_READWRITE	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

ShareMode: Specifies how the profile should be shared, if the profile is contained in a file. A value of zero prevents the profile from being shared at all. The parameter can contain one or both of the following constants (combined by addition or logical OR).

Value	Meaning
FILE_SHARE_READ	Other open operations can be performed on the profile for read access.
FILE_SHARE_WRITE	Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

CreationMode: Specifies which actions to take on the profile while opening it, if it is contained in a file. This parameter must take one of the following constant values.

Value	Meaning
CREATE_NEW	Creates a new profile. Fails if the profile already exists.
CREATE_ALWAYS	Creates a new profile. Overwrites the profile if it exists.
OPEN_EXISTING	Opens the profile. Fails if it does not exist
OPEN_ALWAYS	Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.
TRUNCATE_EXISTING	Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default CAMP and GMMP in the registry.

When OpenColorProfile encounters an ICC profile with an embedded WCS profile, and if the dwType member within the Profile structure does not take the value DONT_USE_EMBEDDED_WCS_PROFILES, it should extract and use the WCS profile(s) contained in this WcsProfilesTag. The HPROFILE returned would be a WCS HPROFILE.

CreationMode flags CREATE_NEW, CREATE_ALWAYS, and TRUNCATE_EXISTING, will always return blank ICC HPROFILEs. If other CreationMode flags are present, InternalOpenColorProfile is called (using the flags as provided by the API) to determine whether the profile is ICC or WCS XML.

Within the ICC code path, an ICC Profile is returned using the requested sharing, access and creation flags as specified in the tables above.

Within the WCS path, the CreationMode flag OPEN_ALWAYS will fail if the profile doesn't exist, since WCS profiles cannot be created or edited within the WCS architecture (they must be edited outside of it, using MSXML6). For the same reason, dwShareMode flag FILE_SHARE_WRITE, and dwDesiredAccess flag PROFILE_READWRITE are ignored within the WCS path.

When the function opens the ICC profile, it will look for a WcsProfilesTag and, if there is one, it will extract and use the original WCS profiles contained therein. (See WcsCreateIccProfile.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default GMMP from the registry. An HPROFILE is a composition of a DMP, CAMP and GMMP.

35.7.16 SetColorProfileHeader(header as WindowsICMProfileHeaderMBS) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The SetColorProfileHeader function sets the header data in a specified ICC color profile.
Notes: Header: the profile header data to write to the specified profile.

Returns true on success.

This function will fail if Profile is not a valid ICC profile.

If the color profile was not opened with read/write permission, SetColorProfileHeader fails.

SetColorProfileHeader overwrites the current header in the ICC profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with and hard coded to ICC tag types and there exist many robust XML parsing libraries.

35.7.17   Properties

35.7.18   ColorProfileHeader as WindowsICMProfileHeaderMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetColorProfileHeader function retrieves or derives ICC header structure from either ICC color profile or WCS XML profile.
Notes: Drivers and applications should assume returning TRUE only indicates that a properly structured header is returned. Each tag will still need to be validated independently using either legacy ICM2 APIs or XML schema APIs.

To determine whether the header is derived from an ICC or DMP profile handle, check the header signature (header bytes 36-39). If the signature is "acsp" (big endian) then an ICC profile was used. If the signature is "cdmp" (big-endian) then a DMP was used.

The distinguishing features that identify a header as having been "synthesized" for a WCS DMP are:

WindowsICMProfileHeaderMBS.Signature = 'pmdc' (little endian = big endian 'cdmp')
WindowsICMProfileHeaderMBS.CMMType = '1scw' (little endian = big endian 'wcs1').
(Read only property)

35.7.19   Handle as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal reference to the object.
Notes: Value is a HPROFILE.
(Read and Write property)

35.7.20   Constants

Creation Mode Constants

Constant	Value	Description
CREATE_ALWAYS	2	Creates a new profile. Overwrites the profile if it exists.
CREATE_NEW	1	Creates a new profile. Fails if the profile already exists.
OPEN_ALWAYS	4	Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.
OPEN_EXISTING	3	Opens the profile. Fails if it does not exist
TRUNCATE_EXISTING	5	Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

Profile sharing flags

Constant	Value	Description
FILE_SHARE_READ	1	Other open operations can be performed on the profile for read access.
FILE_SHARE_WRITE	2	Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

Profile access flags

Constant	Value	Description
PROFILE_READ	1	Opens the profile for read access.
PROFILE_READWRITE	2	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

CreateICCProfile Flags

Constant	Value	Description
WCS_DEFAULT	0	Specifies that the new ICC profile contains the original WCS profile in a private WcsProfilesTag.
WCS_ICCONLY	&h00010000	Specifies that the new ICC profile does not contain either the WcsProfilesTag or the original WCS profile.

35.8   class WindowsICMSetupMBS

35.8.1   class WindowsICMSetupMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: This class is made to query color matching setup information from the user.
Notes: This class contains information that the Setup function uses to initialize the ColorManagement
dialog box. After the user closes the dialog box, Setup returns information about the user's selection in this
class.
Blog Entries

- MBS Xojo Plugins, version 21.6pr3
- MBS Real Studio Plugins, version 11.1pr11

35.8.2   Methods

35.8.3   Setup as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The Setup function creates a Color Management dialog box that lets the user choose whether
to enable color management, and if so, provides control over the color profiles used and over the rendering
intent.
Notes: Returns true if the user clicked OK and false on any error or when dialog was cancelled.

35.8.4   Properties

35.8.5   DisplayName as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string naming the monitor to be used for color management.
Notes: If this is not the name of a valid monitor, the first enumerated monitor is used.
(Read and Write property)

35.8.6   Flags as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A set of bit flags used to initialize the dialog box.
Notes: If set to 0 on entry, all controls assume their default states.

When the dialog box returns, these flags are set to indicate the user's input.

See CMS_* flag constants.
(Read and Write property)

35.8.7 MonitorProfile as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string in which to place the name of the user-selected monitor profile.
Notes: If the CMS_SETMONITORPROFILE flag is used, this flag can also be used to select a profile other than the monitor default when the dialog is first displayed.
(Read and Write property)

35.8.8 Parent as Variant

Plugin Version: 11.1, Platform: Windows, Targets: Desktop only.

Function: The parent window.
Notes: Can reference a Window or DesktopWindow object.
(Read and Write property)

35.8.9 PrinterName as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string naming the printer on which the image is to be rendered.
Notes: If this is not a valid printer name, the default printer is used and named in the dialog.
(Read and Write property)

35.8.10 PrinterProfile as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string in which to place the name of the user-selected printer profile.
Notes: If the CMS_SETPRINTERPROFILE flag is used, this flag can also be used to select a profile other than the printer default when the dialog is first displayed.
(Read and Write property)

35.8.11   ProofingIntent as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The type of color management desired for the proofed image.
Notes: Valid values are:

INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC
INTENT_ABSOLUTE_COLORIMETRIC

For more information, see Rendering Intents.
http://msdn.microsoft.com/en-us/library/dd372183(v=vs.85).aspx
(Read and Write property)

35.8.12   RenderIntent as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The type of color management desired.
Notes: Valid values are:

INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC
INTENT_ABSOLUTE_COLORIMETRIC

For more information, see Rendering Intents.
http://msdn.microsoft.com/en-us/library/dd372183(v=vs.85).aspx
(Read and Write property)

35.8.13   SourceName as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: An application-specified string which describes the source profile of the item for which color
management is to be performed.
Notes: If this is ””, the Image Source control displays the name of the Windows default color profile.
(Read and Write property)

35.8.14 TargetProfile as String

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A string in which to place the name of the user-selected target profile for proofing.
Notes: If the CMS_SETTARGETPROFILE flag is used, this flag can also be used to select a profile other than the printer default when the dialog is first displayed.
(Read and Write property)

35.8.15 Events

35.8.16 Apply

Plugin Version: 11.1, Platform: Windows, Targets: .

Function: This event is invoked when the Apply button of the Color Management dialog box is selected.

35.8.17 Idle

Plugin Version: 11.1, Platform: Windows, Targets: .

Function: This event is called regularly if the CMS_USEHOOK flag is used.

35.8.18 Constants

Flag Constants

Constant	Value	Description
CMS_DISABLEICM	1	If set on entry, this flag indicates that the "Enable Color Management" ch box is cleared, disabling all other controls. If set on exit, it means that the u does not wish color management performed.
CMS_DISABLEINTENT	1024	
CMS_DISABLERENDERINTENT	2048	
CMS_ENABLEPROOFING	2	If set on entry, this flag indicates that the Proofing controls are to be enabl and the Proofing check box is checked. If set on exit, it means that the u wishes to perform color management for a different target device than selected printer.
CMS_SETMONITORPROFILE	16	If set on entry, this flag indicates that the color management profile name the MonitorProfile member is to be the initial selection in the monitor pro control. If the specified profile is not associated with the monitor, this fla ignored, and the default profile for the monitor is used.
CMS_SETPRINTERPROFILE	32	If set on entry, this flag indicates that the color management profile name the PrinterProfile member is to be the initial selection in the printer pro control. If the specified profile is not associated with the printer, this fla ignored, and the default profile for the printer is used.
CMS_SETPROOFINTENT	5	Ignored unless CMS_ENABLEPROOFING is also set. If set on entry, a CMS_ENABLEPROOFING is also set, this flag indicates that the Proof Intent member is to be used to initialize the Target Rendering Intent cont Otherwise, the control defaults to Picture rendering. This flag is set on exi proofing is enabled.
CMS_SETRENDERINTENT	4	If set on entry, this flag indicates that the RenderIntent member contains value to use to initialize the Rendering Intent control. Otherwise, the cont defaults to Picture rendering. This flag is set on exit if WCS is enabled.
CMS_SETTARGETPROFILE	64	If set on entry, this flag indicates that the color profile named in the Targ Profile member is to be the initial selection in the target profile control. If specified profile is not installed, this flag is ignored, and the default profile the printer is used. If the printer has no default profile, then the first profil alphabetical order will be displayed.
CMS_USEAPPLYCALLBACK	256	If set on entry, this flag indicates that the SetupColorMatching function sho call the Apply event.
CMS_USEDESCRIPTION	512	If set on entry, this flag instructs the Setup function to retrieve the pro description contained in the profile description tags (See ICC Profile Form Specification v3.4). It will insert them into the Monitor Profile, Printer Prof Emulated Device Profile edit boxes in the Color Management common dia box.
CMS_USEHOOK	128	The idle event is called regularly if this flag is set.

Rendering Intent Constants

Constant	Value	Description
INTENT_ABSOLUTE_COLORIMETRIC	3	Maintain the white point. Match the colors to their nearest color in nation gamut.
INTENT_PERCEPTUAL	0	Maintain contrast. Used for photographs and natural images.
INTENT_RELATIVE_COLORIMETRIC	1	Maintain colorimetric match. Used for graphic designs and named
INTENT_SATURATION	2	Maintain saturation. Used for business charts and other situation undithered colors are required.

35.9 class WindowsICMTransformMBS

35.9.1 class WindowsICMTransformMBS

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The class for a color transformation.
Blog Entries

- MBS Real Studio Plugins, version 11.1pr11

35.9.2 Methods

35.9.3 CheckColors(InputColors() as WindowsICMColorMBS, ctInput as Integer, Results() as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CheckColors function determines whether the colors in an array lie within the output gamut of a specified transform.
Notes: InputColors: an array of colors.
ctInput: Specifies the input color type.
Results: An array of nColors bytes that receives the results of the test.

If this function succeeds, the return value is TRUE.

If this function fails, the return value is FALSE. For extended error information, call GetLastError.

Remarks

If the input color type is not compatible with the color transform, CheckColors fails.

The function places results of the tests in the array pointed to by paResult. Each byte in the array corresponds to a COLOR element in the array pointed to by paInputColors and has an unsigned value between 0 and 255. The value 0 denotes that the color is in gamut, while a nonzero value denotes that it is out of gamut. For any integer n such that $0 < n < 255$, a result value of n+1 indicates that the corresponding color is at least as far out of gamut as would be indicated by a result value of n.

The out-of-gamut information in the gamut tags created in WCS use the perceptual color distance in CIECAM02, which is the mean square root in CIECAM02 Jab space. The distance in the legacy ICC profile gamut tags is the mean square root in CIELAB space. We recommend that you use the CIECAM02 space when it is available because it provides more perceptually accurate distance metrics.

35.9.4 Constructor(LogColorSpace as WindowsICMLogColorSpaceMBS, Dest-Profile as WindowsICMProfileMBS, TargetProfile as WindowsICMProfileMBS, Flags as Integer)

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateColorTransform function creates a color transform that applications can use to perform color management.
Notes: LogColorSpace: The input colorspace.

DestProfile: The profile of the destination device. The function determines whether the profile contains International Color Consortium (ICC) or Windows Color System (WCS) profile information.

TargetProfile: The profile of the target device. The function determines whether the profile contains ICC or WCS profile information.

Flags: Specifies flags to used control creation of the transform. See Remarks.

On success the handle property is not zero.

If the target profile is nil, the transform goes from the source logical color space to the destination profile. If the target profile is given, the transform goes from the source logical color space to the target profile and then to the destination profile. This allows previewing output meant for the target device on the destination device.

The values in Flags are intended as hints only. The color management module must determine the best way to use them.

Windows Vista: Three new flags have been added that can be used with dwFlags:

PRESERVEBLACK	If this bit is set, the transform engine inserts the appropriate black generation GMMP as the last GMMP in the transform sequence. This flag only works in a pure WCS transform.
SEQUENTIAL_TRANSFORM	If this bit is set, each step in the WCS processing pipeline is performed for every pixel in the image and no optimized color transform is built. This flag only works in a pure WCS transform.

Restrictions: A transform created with the SEQUENTIAL_TRANSFORM flag set may only be used in the thread on which it was created and only for one color translation call at a time. COM must be initialized prior to creating the sequential transform and must remain initialized for the lifetime of the transform object.

For details, see CMM Transform Creation Flags. All of the flags mentioned there are supported for all types

WCS_ALWAYS   If this bit is set, even all-ICC transforms will use the WCS code path.

of transforms, except for FAST_TRANSLATE, which only works in a pure ICC-to-ICC transform.

The CreateColorTransform function is used outside of a device context. Colors may shift when transforming from a color profile to the same color profile. This is due to precision errors. Therefore, a color transform should not be performed under these circumstances.

The B2Ax tags are required for any profile that is the target of a transform.

WCS transform support for ICC ColorSpace profiles is limited to RGB colorspace profiles. The following ICC profile types cannot be used in a CITE-processed transform, either a mixed WCS/ICC transform or an all-ICC transform with WCS_ALWAYS set:

Non-RGB ColorSpace profiles
NamedColor profiles
n-channel profiles (where n >8)
DeviceLink profiles
Abstract profiles
See also:

35.9.5 Constructor(Profiles() as WindowsICMProfileMBS, Intents() as Integer, Flags as Integer, indexPreferredCMM as Integer)

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The CreateMultiProfileTransform function accepts an array of profiles or a single device link profile and creates a color transform that applications can use to perform color mapping.
Notes: Profiles: An array of profiles to be used. The function determines whether the HPROFILEs contain International Color Consortium (ICC) or Windows Color System (WCS) profile information and processes them appropriately. When valid WCS profiles are returned by OpenColorProfile, these profiles contain the combination of DMP, CAMP, and GMMP profiles.
Intent: An array of intents to use. Each intent is one of the following values:

INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC
INTENT_ABSOLUTE_COLORIMETRIC

GMMPs are a generalization of intents. There are two possible sources of intents: the "destination" profile and the intent list parameter to CreateMultiProfileTransform. The term "destination" is not used since all but two of the profiles in the profile list parameter will serve as first destination and then source.

For more information, see Rendering Intents.

nIntents
Specifies the number of elements in the intents array: can either be 1 or the same value as nProfiles. For profile arrays that contain any WCS profiles, the first rendering intent is ignored and only nProfiles -1 elements are used for these profile arrays. The maximum number of nIntents is 10.

Flags: Specifies flags used to control creation of the transform.

indexPreferredCMM: Specifies the one-based index of the color profile that indicates what color management module (CMM) to use. The application developer may allow Windows to choose the CMM by setting this parameter to INDEX_DONT_CARE. See Using Color Management Modules (CMM) Third party CMMs are only available for ICC workflows. Profile arrays containing WCS profiles will ignore this flag. It is also ignored when only ICC profiles are used and when the WCS_ALWAYS flag is used.

On success the handle property is not zero.

If a device link profile is being used, the function will fail if Profiles contains more than one value.

The array of intents specifies how profiles should be combined. The nth intent is used for combining the nth profile in the array. If only one intent is specified, it is used for the first profile, and all other profiles are combined using Match intent.

The values in Flags are intended as hints only. The color management module must determine the best way to use them.

Windows Vista: Three new flags have been added that can be used with dwFlags:

PRESERVEBLACK	If this bit is set, the transform engine inserts the appropriate black generation GMMP as the last GMMP in the transform sequence. This flag only works in a pure WCS transform.
SEQUENTIAL_TRANSFORM	If this bit is set, each step in the WCS processing pipeline is performed for every pixel in the image and no optimized color transform is built. This flag only works in a pure WCS transform.

Restrictions: A transform created with the SEQUENTIAL_TRANSFORM flag set may only be used in the thread on which it was created and only for one color translation call at a time. COM must be initialized

prior to creating the sequential transform and must remain initialized for the lifetime of the transform object.

WCS_ALWAYS   If this bit is set, even all-ICC transforms will use the WCS code path.

For details, see CMM Transform Creation Flags. All of the flags mentioned there are supported for all types of transforms, except for FAST_TRANSLATE and USE_RELATIVE_COLORIMETRIC, which only work in a pure ICC-to-ICC transform.

The CreateMultiProfileTransform function is used outside of a device context. Colors may shift when transforming from a color profile to the same color profile. This is due to precision errors. Therefore, a color transform should not be performed under these circumstances.

We recommend that there be only one GMMP between a source and destination DMP. Gamut boundary descriptions (GBDs) are created from the DMP/CAMP combinations. The subsequent GMMPs use the GDBs prior to them in the processing chain until there exists a DMP/CAMP GBD next in the sequence to be used. For example, assume a sequence DMP1, CAMP1, GMMP1, GMMP2, GMMP3, DMP2, CAMP2, GMMP4, GMMP5, CAMP3, DMP3. Then GMMP1, GMMP2 use GBD1 as their source and destination. Then GMMP3 uses GBD1 as source and GBD2 as destination. Then GMMP4 uses GBD2 as source and destination. Finally GMMP5 uses GBD2 as source and GBD3 as destination. This assumes no GMMP is identical to one next to it.

For WCS profiles, we recommend that the rendering intents be set to DWORD_MAX in order to use the GMMP within the WCS profile handle. This is because the array of rendering intents takes precedence over the rendering intents or gamut mapping models specified or contained in the profiles specified by the PROFILEs. The array of rendering intents references the default GMMP for those rendering intents. Ideally, only one gamut mapping is performed between a source and destination device by setting one or the other GMMP to NULL when creating the HPROFILE with WCS profile information. Any legacy application that uses a WCS DMP will invoke a sequence of GMMPs. GDBs are chosen based on DMPs and CAMPs. For intermediate GMMP gamut boundaries, the source and destination GBDs are used.

In summary, if ubound(Intents)=0, then the first GMM is set based on the GMMP that is set as default* for the padwIntent value, unless that value is DWORD_MAX, in which case the embedded GMM information from the second profile is used (The embedded GMM information is either a GMMP or, in the case of an ICC profile, the baseline GMM corresponding to** the intent from the profile header). The remainder of the GMMs are set based on the GMMP that is set as default* for RelativeColorimetric.

If ubound(Intents) = ubound(Profiles) - 1, then each GMM is set based on the GMMP that is set as default* for the value in the padwIntent array at the corresponding index, except where padwIntent values are DWORD_MAX. For values in the padwIntent array that are DWORD_MAX, the GMMs at corresponding positions are set based on the embedded GMM information from the second of the two profiles whose gamuts are mapped by the GMM. (Again, the embedded GMM information is either a GMMP or, in the case of an ICC profile, the baseline GMM corresponding to** the intent from the profile header).

If ubound(Intents) = ubound(Profiles), then first intent is ignored and funtion behaves as it does in the case when ubound(Intents) = ubound(Profiles) - 1.

Any other combination of padwIntents and nIntents will return an error.

- "set as default" means that the default GMMP is queried using WcsGetDefaultColorProfile with its profileManagementScope parameter set to WCS_PROFILE_MANAGEMENT_SCOPE_CUR-RENT_USER. This may return either current-user or system-wide defaults as described in the documentation for WcsGetDefaultColorProfile.

- "GMM corresponding to" does not mean "GMM from the GMMP set as default for". Instead it means "a constant association between ICC profile intents and baseline GMM algorithms."

WCS transform support for ICC ColorSpace profiles is limited to RGB colorspace profiles. The following ICC profile types cannot be used in a CITE-processed transform, either a mixed WCS/ICC transform or an all-ICC transform with WCS_ALWAYS set:

- Non-RGB ColorSpace profiles
- NamedColor profiles
- n-channel profiles (where n >8)
- DeviceLink profiles
- Abstract profiles

See also:

35.9.6 GetCMMInfo(what as Integer) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The GetCMMInfo function retrieves various information about the color management module (CMM) that created the specified color transform.
Notes: what: Specifies the information to be retrieved. This parameter can take one of the following constant values.

If this function succeeds, the return value is the information specified in What.
If this function fails, the return value is zero.

Value	Meaning
CMM_WIN_VERSION	Retrieves the version of Windows targeted by the color management module (CMM).
CMM_DLL_VERSION	Retrieves the version number of the CMM.
CMM_IDENT	Retrieves the CMM signature registered with the International Color Consortium (ICC).

35.9.7   TranslateBitmapBits(SrcBits as memoryblock, InputType as Integer, Width as Integer, Height as Integer, InputRowBytes as Integer, DestBits as memoryblock, DestType as Integer, DestRowBytes as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The TranslateBitmapBits function translates the colors of a bitmap having a defined format so as to produce another bitmap in a requested format.
Notes: SrcBits: Pointer to the bitmap to translate.
InputType: Specifies the format of the input bitmap. Use one of the BM_* constants.
Width: Specifies the number of pixels per scan line in the input bitmap.
Height: Specifies the number of scan lines in the input bitmap.
InputRowBytes: Specifies the number of bytes from the beginning of one scan line to the beginning of the next in the input bitmap; if set to zero, the function assumes that scan lines are padded so as to be DWORD-aligned.
DestBits: Pointer to the buffer in which to place the translated bitmap.
DestType: Specifies the format of the output bitmap. Use one of the BM_* constants.
DestRowBytes: Specifies the number of bytes from the beginning of one scan line to the beginning of the next in the output bitmap; if set to zero, the function assumes that scan lines should be padded to be DWORD-aligned.

If this function succeeds, the return value is TRUE.
If this function fails, the return value is FALSE.

Remarks

If the input and output formats are not compatible with the color transform, this function fails.

When either of the floating point BMFORMATs, BM_32b_scARGB or BM_32b_scRGB are used, the color data being translated should not contain NaN or infinity. NaN and infinity are not considered to represent legitimate color component values, and the result of translating pixels containing NaN or infinity is meaningless in color terms. NaN or infinity values in the color data being processed will be handled silently, and an error will not be returned.

35.9.8 TranslateColors(InputColors() as WindowsICMColorMBS, ctInput as Integer, OutputColors() as WindowsICMColorMBS, ctOutput as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The TranslateColors function translates an array of colors from the source color space to the destination color space as defined by a color transform.
Notes: InputColors: The input color.
ctInput: Specifies the input color type.
OutputColors: The output color.
ctOutput: Specifies the output color type.

Returns true on success and false on failure.

35.9.9 TranslatePictures(InputPicture as picture, OutputPicture as picture) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The TranslateBitmapBits function translates the colors of a bitmap having a defined format so as to produce another bitmap in a requested format.
Notes: InputPicture: input picture
OutputPicture: output picture

If this function succeeds, the return value is TRUE.
If this function fails, the return value is FALSE.

Make sure the pictures have the same size.

If the input and output formats are not compatible with the color transform, this function fails.

35.9.10 Properties

35.9.11 Handle as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The internal handle for the transformation.
Notes: (Read and Write property)

35.9.12 Events

35.9.13 Progress(Maximum as Integer, Current as Integer) as boolean

Plugin Version: 11.1, Platform: Windows, Targets: .

Function: The progress event for long operations.

35.9.14 Constants

Constants

Constant	Value	Description
CMM_DLL_VERSION	3	One of the values for the GetCMMInfo function. Retrieves the version number of the CMM.
CMM_IDENT	1	One of the values for the GetCMMInfo function. Retrieves the CMM signature registered with the International Color Consortium (ICC).
CMM_WIN_VERSION	0	One of the values for the GetCMMInfo function. Retrieves the version of Windows targeted by the color management module (CMM).
INDEX_DONT_CARE	0	A special value for index in constructor. The application developer may allow Windows to choose the CMM by setting the indexPreferredCMM parameter to INDEX_DONT_CARE for the constructor.
WCS_ALWAYS	&h200000	One of the flags passed when creating a transformation. If this bit is set, even all-ICC transforms will use the WCS code path.

Transform creation Flags

Constant	Value	Description
BEST_MODE	3	Transform will be used for the display of the highest-quality image the target device.
ENABLE_GAMUT_CHECKING	&h10000	Use this transform for gamut checking.
FAST_TRANSLATE	&h40000	Look up color only. Do not interpolate the color.
NORMAL_MODE	2	Transform will be used for normal image display. Average image
PRESERVEBLACK	&h100000	If this bit is set, the transform engine inserts the appropriate blac GMMP as the last GMMP in the transform sequence. This flag o a pure WCS transform.
PROOF_MODE	1	Transform will be used to preview the image. Low image quality.
SEQUENTIAL_TRANSFORM	&h80800000	If this bit is set, each step in the WCS processing pipeline is p every pixel in the image and no optimized color transform is bui only works in a pure WCS transform. Restrictions: A transform created with the SEQUENTIAL_TF flag set may only be used in the thread on which it was created one color translation call at a time. COM must be initialized prio the sequential transform and must remain initialized for the lif transform object.
USE_RELATIVE_COLORIMETRIC	&h20000	

Bitmap Format Constants

Constant	Value	Description
BM_10b_G3CH	&h0404	32 bits per pixel. 10 bits are used for each color channel. The 2 most significa bits are ignored.
BM_10b_Lab	&h0403	32 bits per pixel. 10 bits are used for each color channel. The 2 most significa bits are ignored.
BM_10b_RGB	9	32 bits per pixel. 10 bits are used for each color channel. The 2 most significa bits are ignored.
BM_10b_XYZ	&h0401	32 bits per pixel. 10 bits are used for each color channel. The 2 most significa bits are ignored.
BM_10b_Yxy	&h0402	32 bits per pixel. 10 bits are used for each color channel. The 2 most significa bits are ignored.
BM_16b_G3CH	&h0504	64 bits per pixel. 16 bits are used for the gray-scale value. Each of the thr color channels uses 16 bits.
BM_16b_GRAY	&h0505	64 bits per pixel. 16 bits are used for the gray-scale value. All other bits a ignored.
BM_16b_Lab	&h0503	64 bits per pixel. 16 bits are used for the gray-scale value. Each of the thr color channels uses 16 bits.
BM_16b_RGB	10	64 bits per pixel. 16 bits are used for the gray-scale value. Each of the thr color channels uses 16 bits.
BM_16b_XYZ	&h0501	64 bits per pixel. 16 bits are used for the gray-scale value. Each of the thr color channels uses 16 bits.
BM_16b_Yxy	&h0502	64 bits per pixel. 16 bits are used for the gray-scale value. Each of the thr color channels uses 16 bits.
BM_32b_scARGB	&h0602	128 bits per pixel. 32 bits are used for each color channel, as defined by t IEEE 32-bit floating point standard.
BM_32b_scRGB	&h0601	96 bits per pixel. 32 bits are used for each color channel, as defined by t IEEE 32-bit floating point standard.
BM_565RGB	1	16 bits per pixel. 5 bits are used for red, 6 for green, and 5 for blue.
BM_5CHANNEL	&h0205	40 bits per pixel. 8 bits apiece are used for each channel.
BM_6CHANNEL	&h0206	48 bits per pixel. 8 bits apiece are used for each channel.
BM_7CHANNEL	&h0207	56 bits per pixel. 8 bits apiece are used for each channel.
BM_8CHANNEL	&h0208	64 bits per pixel. 8 bits apiece are used for each channel.
BM_BGRTRIPLETS	4	24 bits per pixel maximum. For three channel colors, such as red, green, a blue, the total size is 24 bits per pixel. For single channel colors, such as gr the total size is 8 bits per pixel.
BM_CMYKQUADS	&h0020	32 bits per pixel. 8 bits are used for each color channel.
BM_G3CHTRIPLETS	&h0204	24 bits per pixel maximum. For three channel values, the total size is 24 b per pixel. For single channel gray scale, the total size is 8 bits per pixel.
BM_GRAY	&h0209	32 bits per pixel. Only the 8 bit gray-scale value is used.
BM_KYMCQUADS	&h0305	32 bits per pixel. 8 bits are used for each color channel.
BM_LabTRIPLETS	&h0203	24 bits per pixel maximum. For three channel, L, a, and b values, the tot size is 24 bits per pixel. For single channel gray scale, the total size is 8 b per pixel.
BM_NAMED_INDEX	&h0405	32 bits per pixel. Named color indices. Index numbering begins at one.
BM_R10G10B10A2	&h0701	Only in Windows Vista.
BM_R10G10B10A2_XR	&h0702	Only in Windows Vista.
BM_R16G16B16A16_FLOAT	&h0703	Only in Windows Vista.
BM_RGBTRIPLETS	2	24 bits per pixel maximum. For three channel colors, such as red, green, a blue, the total size is 24 bits per pixel. For single channel colors, such as gr the total size is 8 bits per pixel.
BM_S2DOT13FIXED_scARGB	&h0604	64 bits per pixel. Color data is stored as one 16-bit word per channel, wit fixed range of -4 to +4, inclusive. A signed format is used, with 1 bit for t sign, 2 bits for the integer portion, and 13 bits for the fractional portion.
BM_S2DOT13FIXED_scRGB	&h0603	48 bits per pixel. Color data is stored as one 16-bit word per channel, wit fixed range of -4 to +4, inclusive. A signed format is used, with 1 bit for t sign, 2 bits for the integer portion, and 13 bits for the fractional portion.
BM_x555G3CH	&h0104	16 bits per pixel. G3CH color space. 5 bits per channel. The most significa bit is ignored.
BM_x555Lab	&h0103	16 bits per pixel. Lab color space. 5 bits per channel. The most significant b

Info selectors for GetInfo

Constant	Value	Description
CMM_DESCRIPTION	5	A text string that describes the color management module.
CMM_DRIVER_VERSION	2	
CMM_FROM_PROFILE	0	
CMM_LOGOICON	6	The logo icon for this CMM.
CMM_VERSION	4	Version of Windows supported.
CMM_WINDOWS_DEFAULT	&h57696E20	

Rendering Intent Constants

Constant	Value	Description
INTENT_ABSOLUTE_COLORIMETRIC	3	Maintain the white point. Match the colors to their nearest color i nation gamut.
INTENT_PERCEPTUAL	0	Maintain contrast. Used for photographs and natural images.
INTENT_RELATIVE_COLORIMETRIC	1	Maintain colorimetric match. Used for graphic designs and named
INTENT_SATURATION	2	Maintain saturation. Used for business charts and other situation undithered colors are required.

Chapter 36

Windows Mutex

36.1   class WindowsMutexMBS

36.1.1   class WindowsMutexMBS

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for a mutex.
Notes: As a mutex must have a unique name you can use this class to find out if an application is allready running.
Try to create a mutex with an unique name and if it allready exists your application was launched twice.
Blog Entries

- New in the MBS Xojo Plugins Version 20.2

- MBS Xojo Plugins, version 20.2pr2

- MBS Plugins 10.3 Release Notes

36.1.2   Methods

36.1.3   close

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources of this object now without waiting for Xojo to do it for you.

36.1.4   Create(name as string)

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new mutex.
Notes: The name must be unique and should not contain a backslash.
Use only ASCII strings for this.
Lasterror is set.

36.1.5   Lock

Plugin Version: 20.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Locks the
Notes: Blocks until we got the lock.
Lasterror is set.

36.1.6   Open(name as string)

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Trys to open a mutex with the given name.
Notes: Fails if the mutex does not exist.
Lasterror is set.

36.1.7   TryLock as Boolean

Plugin Version: 20.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Tries to lock the mutex.
Notes: Returns true if we got the lock or false if not.
Lasterror is set.

36.1.8   Unlock

Plugin Version: 20.2, Platform: Windows, Targets: Desktop, Console & Web.

Function: Unlocks the mutex.
Notes: Lasterror is set.

36.1.9 Properties

36.1.10 Handle as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The handle of the Mutex for use with Declares.
Notes: (Read and Write property)

36.1.11 Lasterror as Integer

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The last error code reported.
Notes: A windows error code.
Or 0 for okay and -1 for "Function not available or parameters wrong."
(Read and Write property)

36.1.12 Name as String

Plugin Version: 3.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: The name used to create or open the mutex.
Notes: (Read only property)

Chapter 37

Windows Registry

37.1 class RegistryFileTypeMBS

37.1.1 class RegistryFileTypeMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: An easy way to register an document icon for Windows.
Notes: You set up the properties of this class and than you call create.
I suggest that you do it like QuickTime or Winzip. Ask the user on first run if he want's your application to register its file types.
And also provide two buttons in the Preferencesdialog to register or unregister the types.

This class works on Windows 7 only if you run the app as administrator.

This function may fail to run if permissions are denied.
Or it may only affect the shadow registry used by Windows to protect itself from having unauthorized apps editing the real registry.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr5

- Plugin notes

37.1.2 Methods

37.1.3 Create as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates this file type.
Example:

dim r as RegistryFileTypeMBS
' set properties for r
dim errorcode as Integer
errorcode=r.create

Notes: Returns:

-1	- Run on Mac OS
0	- OK
1	- Failed to create main extension key.
2	- Failed to create the description key for file type.
3	- Failed to create the key for the icon.
4	- Failed to create the key for the open description.
5	- Required parameters not provided.

This function may fail to run if permissions are denied.
Or it may only affect the shadow registry used by Windows to protect itself from having unauthorized apps editing the real registry.

37.1.4   Remove as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes this file type.
Example:

dim r as RegistryFileTypeMBS
dim errorcode as Integer

errorcode=r.remove

Notes: Returns:

Note: The return values changed in MBS Plugin 3.2 to now return Windows error codes!

This function may fail to run if permissions are denied.

-1   - Run on Mac OS or parameters bad.

0   - OK

x   - A Windows error code like 5 for permissions denied.

Or it may only affect the shadow registry used by Windows to protect itself from having unauthorized apps editing the real registry.

37.1.5 Properties

37.1.6 AppFile as FolderItem

Platform: Windows, Targets: Desktop, Console & Web.

Function: The application to launch to open the file.
Example:

dim r as new RegistryFileTypeMBS
r.Appfile=app.applicationfilembs // it is my file ;-)

Notes: (Read and Write property)

37.1.7 Description as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The description for this file type.
Example:

dim r as new RegistryFileTypeMBS
r.description="Monkeybread File"

Notes: The text to show in the explorer in column view.
(Read and Write property)

37.1.8 Extension as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The extension for this file type.
Example:

dim r as new RegistryFileTypeMBS

r.extension=".MBS"

Notes: The extension should start with a dot follow by 3 uppercase letters and it should be unique. But as there are more file types than combinations of 3 letters this may be difficult...
(Read and Write property)

37.1.9   FileType as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: A internal unique signature for this file type.
Example:

dim r as new RegistryFileTypeMBS

r.FileType="MBSFile"

Notes: May contain any character except chr(0) and should be ANSI encoded.
(Read and Write property)

37.1.10   Iconfile as FolderItem

Platform: Windows, Targets: Desktop, Console & Web.

Function: The file with the icon.
Example:

dim r as new RegistryFileTypeMBS

r.iconfile=app.applicationfileMBS // it is my file ;-)

Notes: You can get the icon out of your application (ID 0 is the default icon).
But you can also use DLL files like "Shell32.dll" for some nice icons.
Last but not least you can use an ICO file which Iconographer can create for you from your Mac Icons.
(Read and Write property)

37.1.11   IconID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID of the Icon to use.
Example:

dim r as new RegistryFileTypeMBS

r.iconid=3

Notes: Your icon file may contain more than one icon. Here you can specify which one to use. First is ID 0.
(Read and Write property)

37.1.12   OpenDescription as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The text to show in the context menu for Open.
Example:

dim r as new RegistryFileTypeMBS

r.OpenDescription="Open MBS file"

Notes: If you don't provide an OpenDescription you'll get "Open".
Here you can describe the default action.
e.g. Stuffit Expander could say "Expand file".
(Read and Write property)

37.2 class RegistryKeyMBS

37.2.1 class RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you access to a Windows Registry Key.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 17.4
- MBS Xojo Plugins, version 17.4pr6
- MBS REALbasic Plugins Version 10.4 release notes
- MBS REALbasic Plugins, version 10.4pr11

37.2.2 Methods

37.2.3 CopyTree(keyname as string, Dest as RegistryKeyMBS) as boolean

Plugin Version: 17.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Copies the specified registry key, along with its values and subkeys, to the specified destination key.
Notes: Returns true on success and false on failure.

37.2.4 CreateKey(name as string, Use64bitRegistry as boolean = false) as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a new subkey.
Notes: If the key is allready present it is just opened.
Returns a registry key or nil on any error.

Use64bitRegistry: Pass true to create key in 64 bit view of the Registry instead of 32 bit view.

37.2.5 Delete(keyname as string) as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes the specified subkey key.
Notes: Windows 95: The Delete function deletes a subkey and all its descendants.

Windows NT: The Delete function deletes the specified subkey. The subkey to be deleted must not have subkeys.

With plugin version 3.2 the NT behavior is worked around to match the Windows 95 behavior. So all sub keys are deleted!

37.2.6 DeleteTree(keyname as string) as boolean

Plugin Version: 17.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes the subkeys and values of the specified key recursively.
Example:

```
dim t as RegistryKeyMBS = RegistryMBS.CurrentUser

// create some keys
dim n as RegistryKeyMBS = t.CreateKey("Hello")
dim w as RegistryKeyMBS = n.CreateKey("World")


w.Value("Test").asString = "Hello"

break // see in regedit

// delete it
if t.DeleteTree("Hello") then
MsgBox "deleted. ok"
else
MsgBox "delete failed"
end if
```

Notes: Returns true on success and false on failure.

37.2.7 Flush

Platform: Windows, Targets: Desktop, Console & Web.

Function: Writes all the attributes of the specified open key into the RegistryMBS.

37.2.8   Item(index as Integer) as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the key at the specified index.
Notes: May return nil on any error like missing access rights.
Some keys can't be opened, but ItemName() may return the name of the key.
See also:

37.2.9   Item(name as string) as RegistryKeyMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the key with the specified name.
Notes: May return nil on any error like missing access rights.
Some keys can't be opened, but ItemName() may return the name of the key.
See also:

37.2.10   ItemName(index as Integer) as string

Plugin Version: 6.3, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the name of the key at the specified index.
Notes: May return "" on any error.

37.2.11   Value(index as Integer) as RegistryValueMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value at the specified index.
See also:

37.2.12   Value(name as string) as RegistryValueMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value item with the specified name.
Notes: The value must not exists. So use this function to make a new key.

Use an empty name (e.g. "") for the default key value.

This method was named ValueItem in plugin version before 10.4.
See also:

37.2.13   ValueName(index as Integer) as string

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the name of the value with the given index.
Notes: May return "" on any error.

37.2.14   Properties

37.2.15   ItemCount as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the count of subkeys.
Example:

```
dim i,c as Integer
dim key as RegistryKeyMBS // the registry item

c=key.ItemCount-1
for i=0 to c
// do something
next
```

Notes: This property calls a System function to get the value. So save it in a local variable instead of calling it in a for loop. See the example.
(Read only property)

37.2.16   name as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the name of the Key.
Notes: (Read only property)

37.2.17 ValueCount as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the count of values.
Example:

dim i,c as Integer
dim key as RegistryKeyMBS

c=key.ValueCount
for i=1 to c
// do something
next

Notes: This property calls a System function to get the value. So save it in a local variable instead of calling it in a for loop. See the example.
(Read only property)

37.3 class RegistryMBS

37.3.1 class RegistryMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you access to the Windows Registry.
Notes: Please send me an email if you need some functions which are now not included.
Blog Entries

- MBS Real Studio Plugins, version 12.5pr10

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr11

37.3.2 Methods

37.3.3 classesRoot as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the ClassesRoot Tree.

37.3.4 CurrentConfig as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the CurrentConfig Tree.

37.3.5 CurrentUser as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the CurrentUser Tree.

37.3.6 getBinaryValue(keypath as string,valuename as string, Use64bitRegistry as boolean = false) as Memoryblock

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you a Memoryblock of the value called valuename in the key found at keypath.
Example:

```
const path="HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion"
dim r as new RegistryMBS
msgBox "This OS is: "+r.getBinaryValue(path, "ProductName").cstring(0)
```

Notes: Returns nil on any error.
Use64bitRegistry: Pass true to create key in 64 bit view of the Registry instead of 32 bit view.

37.3.7   getStringValue(keypath as string,valuename as string, Use64bitRegistry as boolean = false) as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you a String of the value called valuename in the key found at keypath.
Example:

```
const path="HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion"
dim r as new RegistryMBS
msgBox "This OS is: "+r.getStringValue(path, "ProductName")
```

Notes: Returns "" on any error.
Use64bitRegistry: Pass true to create key in 64 bit view of the Registry instead of 32 bit view.

37.3.8   Key(keypath as string, Use64bitRegistry as boolean = false) as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the given path of a key.
Example:

```
const path="HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion"
dim r as new RegistryMBS
msgBox r.Key(path).Name
```

Notes: If you need you can recreate this function in Xojo code to have some error checking. This function here will return nil on any error.

Use64bitRegistry: Pass true to create key in 64 bit view of the Registry instead of 32 bit view.

37.3.9   LocalMachine as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the LocalMachine Tree.

37.3.10   PerformanceData as RegistryKeyMBS

Plugin Version: 10.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the PerformanceData Tree.

37.3.11   Users as RegistryKeyMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you an RegistryKeyMBS object for the Users Tree.

37.4   class RegistryValueMBS

37.4.1   class RegistryValueMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: Gives you access to a value of a RegistryMBS Key.
Blog Entries

- MBS Xojo Plugins, version 18.3pr6

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr11

37.4.2   Methods

37.4.3   Delete as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Deletes this value.
Notes: The RegistryValueMBS object should be destroyed after deleting.

37.4.4   SetBinaryMem(typ as Integer,data as Memoryblock)

Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the value to the content of the memoryblock.
Notes: Type may be one of this values:

37.4.5   SetBinaryStr(typ as Integer,data as String)

Platform: Windows, Targets: Desktop, Console & Web.

Function: Sets the value to the content of the string.
Notes: Type may be one of this values:

0	No value type
1	Nul terminated string
2	Nul terminated string (with environment variable references)
3	Free form binary
4	32-bit number (LittleEndian)
5	32-bit number (BigEndian)
6	Symbolic Link (unicode)
7	Multiple Unicode strings
8	Resource list in the resource map
9	Resource list in the hardware description
10	Resource requirements list
11	64-bit number

37.4.6   Properties

37.4.7   asBinary as Memoryblock

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value as a memoryblock.
Notes: Can also be used to set the value.
(Read and Write property)

37.4.8   asBinaryString as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value as a string with binary content.
Notes: This string may include chr(0).
Can also be used to set the value.
(Read and Write property)

37.4.9   asLong32 as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value as a 32bit integer.
Notes: Can also be used to set the value.
(Read and Write property)

1386   CHAPTER 37. WINDOWS REGISTRY

0	No value type
1	Nul terminated string
2	Nul terminated string (with environment variable references)
3	Free form binary
4	32-bit number (LittleEndian)
5	32-bit number (BigEndian)
6	Symbolic Link (unicode)
7	Multiple Unicode strings
8	Resource list in the resource map
9	Resource list in the hardware description
10	Resource requirements list
11	64-bit number

37.4.10   asLong64 as Int64

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value as a 64bit integer.
Example:

```
dim v as RegistryValueMBS
v.aslong64=v.aslong64+1 // add one
```

Notes: Can also be used to set the value.
(Read and Write property)

37.4.11   asString as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the value as a string.
Notes: Can also be used to set the value.
(Read and Write property)

37.4.12   isBinary as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the value is binary data.

Notes: (Read only property)

37.4.13 isLong32 as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the value is a 32 bit integer.
Notes: (Read only property)

37.4.14 isLong64 as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the value is a 64 bit Integer.
Notes: (Read only property)

37.4.15 isString as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns true if the value is a string.
Notes: (Read only property)

37.4.16 name as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the name of the Value.
Notes: (Read only property)

37.4.17 size as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the size of this value.
Notes: (Read only property)

37.4.18   type as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: Returns the type of the value.
Notes: Type may be one of this values:

0	No value type
1	Nul terminated string
2	Nul terminated string (with environment variable references)
3	Free form binary
4	32-bit number (LittleEndian)
5	32-bit number (BigEndian)
6	Symbolic Link (unicode)
7	Multiple Unicode strings
8	Resource list in the resource map
9	Resource list in the hardware description
10	Resource requirements list
11	64-bit number

(Read only property)

Chapter 38

Windows Shortcuts

38.1 class WindowsInternetShortcutMBS

38.1.1 class WindowsInternetShortcutMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to create internet shortcuts on Windows.
Example:

#if targetwin32
dim w as WindowsInternetShortcutMBS

w=new WindowsInternetShortcutMBS

w.URL="http://www.monkeybreadsoftware.de"
w.Command=3
w.Icon=windowsSystemFolder.child("shell32.dll")
w.iconID=41
w.Location=SpecialFolder.Desktop.child("A new link to explorer.lnk")
w.Working=volume(0)

if w.CreateInternetShortCut then
if w.AddInternetShortCutIcon then
msgBox "Shortcut created."
else
msgBox "Failed to add icon."
end if
else
msgBox "Failed to create shortcut."
end if
#endif

38.1.2   Methods

38.1.3   CreateInternetShortCut as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a shortcut on the specified location.
Notes: Version 10.1: Now adds the icon, too.

38.1.4   Properties

38.1.5   Command as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The code for the showing of the application.
Notes: Use e.g. 3 for full screen.
(Read and Write property)

38.1.6   Icon as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The location of the icon file.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

38.1.7   IconID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID of the icon inside the icon file.
Notes: (Read and Write property)

38.1.8   Location as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The location where the shortcut is created or modified.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

38.1.9   url as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The URL where the shortcut will point to.
Notes: (Read and Write property)

38.1.10   WorkingDirectory as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The working directory for the browser.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

38.2 class WindowsShortCutMBS

38.2.1 class WindowsShortCutMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to create shortcuts on Windows.
Example:

dim w as WindowsShortCutMBS

w = new WindowsShortCutMBS

w.Arguments = ""
w.Command = 3
w.Icon = specialfolder.system.child("shell32.dll").NativePath
w.iconID = 41
w.Location = SpecialFolder.Desktop.child("A new link to explorer.lnk").NativePath
w.Target = SpecialFolder.Windows.child("explorer.exe").NativePath
w.WorkingDirectory = volume(0).NativePath

if w.CreateShortCut then
msgBox "Shortcut created."
else
msgBox "Failed to create shortcut."
end if

Blog Entries

- MBS Xojo Plugins, version 21.6pr3

38.2.2 Methods

38.2.3 CreateShortCut as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a shortcut on the specified location.
Notes: Returns true on success and false on any error.
Version 10.1: Now adds the icon, too.

38.2.4 ResolveShortCut(DisableGUI as boolean=false, DisableSearch as boolean=false) as boolean

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Resolves the shortcut.
Notes: Fills the properties target, description, Argument, WorkingDirectory, Icon and IconID.
Returns false on any error.

Parameters added in plugin version 7.4:

DisableGUI:
Do not display a dialog box if the link cannot be resolved. The time-out duration will be set to the default value of 3,000 milliseconds (3 seconds).

DisableSearch:
Do not execute the search heuristics.

For more details check this page in Microsoft's documentation:
http://msdn.microsoft.com/en-us/library/bb774952(v=vs.85).aspx

38.2.5 Properties

38.2.6 Arguments as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: Arguments to the application which is linked to the shortcut.
Notes: (Read and Write property)

38.2.7 Command as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The code for the showing of the application.
Notes: Constants which may be useful:

(Read and Write property)

SW_HIDE	= 0
SW_NORMAL	= 1
SW_SHOWMINIMIZED	= 2
SW_SHOWMAXIMIZED	= 3
SW_MAXIMIZE	= 3
SW_SHOWNOACTIVATE	= 4
SW_SHOW	= 5
SW_MINIMIZE	= 6
SW_SHOWMINNOACTIVE	= 7
SW_SHOWNA	= 8
SW_RESTORE	= 9
SW_SHOWDEFAULT	= 10
SW_FORCEMINIMIZE	= 11

38.2.8   Description as String

Plugin Version: 3.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The description of the shortcut.
Notes: (Read and Write property)

38.2.9   Icon as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The location of the icon file.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

38.2.10   IconID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: The ID of the icon inside the icon file.
Notes: (Read and Write property)

38.2.11   Location as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The location where the shortcut is created or modified.
Notes: (Changed from FolderItem to String in plugin version 3.4)

(Read and Write property)

38.2.12 ParentWindow as Variant

Plugin Version: 3.4, Platform: Windows, Targets: Desktop only.

Function: The parent window used when a dialog must be opened.
Notes: Can reference a Window or DesktopWindow object.
(Read and Write property)

38.2.13 Target as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The target where the shortcut will point to.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

38.2.14 WorkingDirectory as String

Platform: Windows, Targets: Desktop, Console & Web.

Function: The working directory of this shortcut.
Notes: (Changed from FolderItem to String in plugin version 3.4)
(Read and Write property)

Chapter 39

Windows System Tray

39.1 Globals

39.1.1 HIconFromFileMBS(IconFile as FolderItem, IconID as Integer) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a Windows Icon Handle for an icon loaded fromt he given icon file.
Notes: What image depths are used and supported depends on the Windows version.
IconID = 0 points to the first icon in the file.
Returns 0 on failure.

The icon file can be .ico, .exe, .dll or similar files with windows icon resource.
This function is useful for declares where you have to pass a HICON parameter.
Blog Entries

- MBS Plugins 11.1 Release notes

- MBS REALbasic Plugins, version 11.1pr2

39.1.2 HIconFromPicturesMBS(Icon as picture, Mask as picture) as Integer

Plugin Version: 11.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: Creates a Windows Icon Handle with the given pictures.
Notes: What image depths are used and supported depends on the Windows version.
Returns 0 on failure.
This function is useful for declares where you have to pass a HICON parameter.
Blog Entries

- MBS Plugins 11.1 Release notes

- MBS REALbasic Plugins, version 11.1pr2

39.2   class WindowsSystemTrayMBS

39.2.1   class WindowsSystemTrayMBS

Platform: Windows, Targets: Desktop, Console & Web.

Function: A class to create an item in the system tray.
Blog Entries

- MBS Real Studio Plugins, version 11.3pr10

- MBS REALbasic Plugins Version 10.4 release notes

- MBS REALbasic Plugins, version 10.4pr8

39.2.2   Methods

39.2.3   Add as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Adds the item to the system tray.
Example:

dim e as WindowsSystemTrayMBS // your system tray object

if e.add then
list.addrow "Added icon."
else
list.addrow "Failed to add icon."
end if

Notes: Returns true if successfull.
You can change the Icon&Mask or the Tooltip string.

39.2.4   Available as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the system tray functions are available.
Notes: Returns true on Windows 95 and newer.
Returns false on Mac OS and Linux.

39.2.5   Modify as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Modifies the item to the system tray.
Notes: Returns true if successfull.
You can change the Icon&Mask or the Tooltip string.

39.2.6   Remove as boolean

Platform: Windows, Targets: Desktop, Console & Web.

Function: Removes the item from the system tray.
Notes: Returns true if successfull.

39.2.7   SetFocus as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: This method resets focus back to the system try area.
Notes: May be useful to call as cleanup after some context menu was shown.

39.2.8   SetIconFile(IconFile as FolderItem, IconID as Integer) as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Loads an icon from the given file.
Example:

dim wst as WindowsSystemTrayMBS // your system tray

call wst.SetIconFile(file,1)

Notes: Returns true on success and false on failure.
What image depths are used and supported depends on the Windows version.

IconID = 0 points to the first icon in the file.

The icon file can be .ico, .exe, .dll or similar files with windows icon resource.

39.2.9   SetIconPicture(Icon as picture, Mask as picture) as boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Loads the icon from the given picture with mask.
Example:

dim myicon as picture // your picture
dim mymask as picture // the mask for the picture
dim wst as WindowsSystemTrayMBS // your system tray

call wst.SetIconPicture(myicon, mymask)

Notes: Returns true on success and false on failure.
What image depths are used and supported depends on the Windows version.

Type is 0 for a small icon and 1 for a big icon.

39.2.10   Properties

39.2.11   BalloonMode as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The balloon mode.
Example:

dim s as WindowsSystemTrayMBS

s.BalloonText = "The text for the balloon."
s.BalloonTitle = "The title for the balloon"
s.BalloonMode = 0 // Error mode

Notes: 0 is Error, 1 is Info and 2 is warning.
(Read and Write property)

39.2.12   BalloonText as string

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The balloon text to show.
Example:

dim wst as WindowsSystemTrayMBS // your system tray object

wst.BalloonTitle="Warning"
wst.BalloonText="You have not clicked on me the last 10 minutes."
call wst.modify

Notes: If BalloonTitle or BalloonText are not empty the balloon is shown when you call modify.
(Read and Write property)

39.2.13   BalloonTimeout as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The balloon timeout in milliseconds.
Example:

dim s as WindowsSystemTrayMBS

s.BalloonText = "The text for the balloon."
s.BalloonTitle = "The title for the balloon"
s.BalloonMode = 0 // Error mode
s.BalloonTimeout = 30000

Notes: A value between 10000 and 30000.
Note that Timeout is valid only in Windows 2000 and Windows XP.
(Read and Write property)
See also:

- 39.2.22 BalloonTimeout(id as Integer, MouseX as Integer, MouseY as Integer)   1403

39.2.14   BalloonTitle as string

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The title of the balloon tooltip.
Example:

dim s as WindowsSystemTrayMBS

s.BalloonText = "The text for the balloon."
s.BalloonTitle = "The title for the balloon"
s.BalloonMode = 0 // Error mode

Notes: If BalloonTitle or BalloonText are not empty the balloon is shown when you call modify.
(Read and Write property)

39.2.15   IconHandle as Integer

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: The icon handle currently used for the tray object.
Notes: zero if no handle is there.
(Read and Write property)

39.2.16   ID as Integer

Platform: Windows, Targets: Desktop, Console & Web.

Function: If you have more than one icon in the system tray, use this ID to see which one is referred to inside an event.
Notes: (Read and Write property)

39.2.17   Tooltip as string

Platform: Windows, Targets: Desktop, Console & Web.

Function: The help string for the tooltip.
Example:

dim e as WindowsSystemTrayMBS // global

e=new WindowsSystemTrayMBS
e.ID=12345678
e.Tooltip="some text"
e.BalloonText="some text"
e.BalloonTitle="some text"
e.BalloonMode=0

Notes: Limited to 63 characters.
(Read and Write property)

39.2.18 UsingNewEvents as Boolean

Plugin Version: 7.4, Platform: Windows, Targets: Desktop, Console & Web.

Function: Whether the new events are enabled.
Notes: The plugin tries to enable new events if possible. That will only work on Windows 2000 or newer.

New events are:
BalloonTimeout
PopupOpen
ContextMenu
BalloonHide
KeySelected
BalloonUserClick
BalloonShow
Selected
(Read only property)

39.2.19 Events

39.2.20 BalloonHide(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The balloon has been hidden.

39.2.21 BalloonShow(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The balloon was shown.
Notes: You will get one balloon event when it disappears: BalloonHide, BalloonUserClick or BalloonTimeout.

39.2.22 BalloonTimeout(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The balloon was removed because of a timeout.
See also:

- 39.2.13 BalloonTimeout as Integer   1401

39.2.23   BalloonUserClick(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The user clicked on the balloon to make it away.

39.2.24   ContextMenu(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: You should show a context menu.

39.2.25   KeySelected(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The system tray item was selected by the keyboard.

39.2.26   MouseLeftButtonDoubleClick(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the left mouse button is pressed for a double click on the system tray icon.

39.2.27   MouseLeftButtonDown(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the left mouse button is pressed on the system tray icon.

39.2.28 MouseLeftButtonUp(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the left mouse button is released on the system tray icon.

39.2.29 MouseMiddleButtonDoubleClick(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the middle mouse button is pressed for a double click on the system tray icon.

39.2.30 MouseMiddleButtonDown(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the middle mouse button is pressed on the system tray icon.

39.2.31 MouseMiddleButtonUp(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the middle mouse button is released on the system tray icon.

39.2.32 MouseMove(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the mouse is moved over the system tray icon.

39.2.33 MouseRightButtonDoubleClick(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the right mouse button is pressed for a double click on the system tray icon.

39.2.34   MouseRightButtonDown(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the right mouse button is pressed on the system tray icon.

39.2.35   MouseRightButtonUp(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: Called if the right mouse button is pressed on the system tray icon.

39.2.36   PopupOpen(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: You should show a popup menu on the mouse position.

39.2.37   Selected(id as Integer, MouseX as Integer, MouseY as Integer)

Plugin Version: 7.4, Platform: Windows, Targets: .

Function: The system tray item was selected.

Chapter 40

Windows Taskbar State

40.1 class WindowsTaskbarStateMBS

40.1.1 class WindowsTaskbarStateMBS

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: A class for information about the settings for the Windows taskbar.
Blog Entries

- MBS Real Studio Plugins, version 12.2pr6

40.1.2 Properties

40.1.3 AlwaysOnTop as Boolean

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: True if the taskbar should be always on the top.
Notes: Value is false on any error.
Settable in plugin version 12.2 and newer.
(Read and Write property)

40.1.4 AutoHide as Boolean

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: True if the taskbar automatically hides itself.
Notes: Value is false on any error.

Settable in plugin version 12.2 and newer.
(Read and Write property)

40.1.5   Bottom as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The bottom value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

40.1.6   Height as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The height value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

40.1.7   Left as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The left value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

40.1.8   Right as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The right value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

40.1.9   Top as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The top value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

40.1.10   Width as Integer

Plugin Version: 3.1, Platform: Windows, Targets: Desktop, Console & Web.

Function: The width value of the taskbar dimensions.
Notes: Value is 0 on any error.
(Read and Write property)

Chapter 41

List of Questions in the FAQ

Chapter 42

The FAQ

42.0.1 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)

42.0.2 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.

42.0.3 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:

42.0.4 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:

42.0.5   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:

42.0.6   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:

42.0.7   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:

42.0.8 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:

42.0.9   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:

42.0.10   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
```

42.0.11   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.

42.0.12   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

42.0.13   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.

42.0.14 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

42.0.15   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.

42.0.16   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.

42.0.17   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

42.0.18 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.

42.0.19 Can I use your ChartDirector plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our ChartDirector plugin works just fine on the Xojo Web Edition.
Example:

```
// The data for the pie chart
dim data(-1) as Double=array(55.0, 18.0, 25.0, 22.0, 18.0, 30.0, 35.0)

// The labels for the pie chart, Words are choosen random to check font!
dim labels(-1) as string=array("Germany","Italy","France","Spain","UK","Poland","Russia")

// The colors to use for the sectors
dim colors(-1) as Integer

colors.Append &h66aaee
colors.Append &heebb22
colors.Append &hbbbbbb
colors.Append &h8844ff

if TargetLinux then
CDBaseChartMBS.SetFontSearchPath "/usr/share/fonts/truetype/msttcorefonts"
end if

// Create a PieChart object of size 360 x 300 pixels
dim c as new CDPieChartMBS(700, 600)
```

c.setBackground(c.linearGradientColor(0, 0, 0, c.getHeight(), &h0000cc, &h000044))
c.setRoundedFrame(&hffffff, 16)
dim tt as CDTextBoxMBS = c.addTitle("ChartDirector Demonstration", "timesbi.ttf", 18)
tt.setMargin(0, 0, 16, 0)
tt.setFontColor(&hFFFFFF)

// Set the center of the pie at (180, 140) and the radius to 100 pixels
c.setPieSize 350,300,150
// Set the sector colors
c.setColors(c.kDataColor, colors)

// Draw the pie in 3D with a pie thickness of 20 pixels
c.set3D(20)

dim t as CDTextBoxMBS = c.setLabelStyle("arialbd.ttf", 10, &h000000)
t.setBackground(CDPieChartMBS.kSameAsMainColor, CDPieChartMBS.kTransparent, CDPieChartMBS.soft-
Lighting(CDPieChartMBS.kRight, 0))
t.setRoundedCorners(8)

// Use local gradient shading for the sectors, with 5 pixels wide
// semi-transparent white (bbffffff) borders
c.setSectorStyle(CDPieChartMBS.kLocalGradientShading, &hbbffffff, 0)

// Set the pie data and the pie labels
c.setData data,labels
call c.setLabelStyle "arialbd.ttf",18

dim pic as picture = c.makeChartPicture
dim wp as new WebPicture(pic, Picture.FormatJPEG) // JPEG makes it smaller and faster

ImageView1.Picture=wp

Notes: Be aware that our plugin produces pictures for you, which you assign to ImageViews. Tranferring 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.

42.0.20   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.

42.0.21   Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

42.0.22   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.

42.0.23   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.

42.0.24 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 AAR-RGGBB, 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

Constant	Value	Description
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

42.0.25   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 "Gill-Sans.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.

42.0.26 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_im-
age_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.

42.0.27 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.

42.0.28   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

42.0.29   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.

42.0.30 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)

42.0.31   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.

42.0.32   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.

42.0.33   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.

42.0.34 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
```

42.0.35 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.

42.0.36   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.

42.0.37   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.

42.0.38 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.

42.0.39 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.

42.0.40 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.

42.0.41 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.

42.0.42 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.

42.0.43   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, appSig-
nature as OSType, nativeTypes as Ptr) as Short Inline68K("701CABFC")
Declare Function GetDocumentKindString Lib "Carbon" (docVRefNum as Short, docType as OSType, doc-
Creator 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.

42.0.44   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

42.0.45   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.

42.0.46   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

42.0.47   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.

42.0.48 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.

42.0.49   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
```

42.0.50   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.

42.0.51   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.

42.0.52 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.

42.0.53   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.

42.0.54   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.

42.0.55   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.

42.0.56   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.

42.0.57   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.

42.0.58 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.

42.0.59 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.

42.0.60 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).

42.0.61 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.

42.0.62 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
```

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.

42.0.63   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>
```

42.0.64   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/NSAutorelease-Pool_Class/Reference/Reference.html

42.0.65 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.

42.0.66 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)

42.0.67   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.

42.0.68   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.

42.0.69 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/ .

42.0.70 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.

42.0.71 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.

42.0.72   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
```

42.0.73   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.

42.0.74   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.

42.0.75   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.

42.0.76   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.

42.0.77   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.

42.0.78   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
```

42.0.79   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
```

42.0.80   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
```

42.0.81 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.

42.0.82   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.

42.0.83   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.

42.0.84   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

42.0.85   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.

42.0.86   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.

42.0.87   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.

42.0.88   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.

42.0.89   How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

42.0.90   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("AQAAAAAAAAAAAAAAAACAFRDRVIAAAABAAAAAAAAABUQ0lQAAAAA
r.AddResource(data,"drag",128,"") 'ditto
r.Close
```

Notes: In general Apple has deprecated this, but a few application still support clippings.

42.0.91 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.

42.0.92 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.

42.0.93 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.

42.0.94   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)

42.0.95   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.

42.0.96   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".

42.0.97   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)

42.0.98   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.

42.0.99   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.

42.0.100   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.

42.0.101 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
```

42.0.102 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.

42.0.103 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.

42.0.104   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.

42.0.105   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

42.0.106   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.

42.0.107   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.

42.0.108   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.

42.0.109   How to extract text from HTML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use both RemoveHTMLTagsMBS and DecodingFromHTMLMBS like this:
Example:

```
dim html as string = ”<p><B>Gr&uuml;&szlig;e</B></P>”
dim htmltext as string = RemoveHTMLTagsMBS(html)
dim text as string = DecodingFromHTMLMBS(htmltext)

MsgBox text // shows: Gr√º√üe
```

Notes: You can use it together with RemoveHTMLTagsMBS to remove html tags. What you get will be the text without tags.
DecodingFromHTMLMBS turns HTML escapes back to unicode characters. Like ä to √§.

42.0.110   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
```

42.0.111   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

42.0.112   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.

42.0.113 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".

42.0.114 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/

42.0.115 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.

42.0.116   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
```

42.0.117   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
```

42.0.118   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.

42.0.119   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

42.0.120   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.

42.0.121 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.

42.0.122   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.

42.0.123   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

42.0.124   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.

42.0.125   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.

42.0.126   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" ...

42.0.127   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.

42.0.128 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
```

42.0.129   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

42.0.130 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.

42.0.131 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

42.0.132   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
```

42.0.133   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.

42.0.134 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.

42.0.135 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!

42.0.136   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

42.0.137   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.

42.0.138   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.

42.0.139 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.

42.0.140   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.

42.0.141 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.

42.0.142 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

42.0.143   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).

42.0.144   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.

42.0.145   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.

42.0.146   How to insert image to NSTextView or TextArea?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to insert file:
Example:

// insert a file to textview

Public Sub InsertFile(textview as NSTextViewMBS, f as FolderItem)
// read to file

```
dim b as BinaryStream = BinaryStream.Open(f)
dim s as string = b.Read(b.Length)

// build wrapper
dim fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
dim fileAttachment as new NSTextAttachmentMBS(fileWrapper)
dim attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttach-
ment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS
method.

42.0.147   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
```

42.0.148   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

42.0.149   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.

42.0.150   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.

42.0.151   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.

42.0.152   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.

42.0.153   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

42.0.154   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.

42.0.155   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.

42.0.156   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.

42.0.157   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
```

42.0.158   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

42.0.159   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

42.0.160   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

42.0.161   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.

42.0.162   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?

42.0.163   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 OS-
Type, 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))
```

42.0.164   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.

42.0.165   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.kFSFile-
OperationDefaultOptions)
```

```
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.

42.0.166 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)

42.0.167 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.

42.0.168 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.

42.0.169 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.

42.0.170 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

42.0.171 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.

42.0.172   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
```

42.0.173   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.

42.0.174   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.

42.0.175   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.

42.0.176   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)+";")

42.0.177   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

42.0.178   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.

42.0.179 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.

42.0.180   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.

42.0.181   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.

42.0.182   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.

42.0.183   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.

42.0.184   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
```

42.0.185   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.

42.0.186   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.

42.0.187   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
```

42.0.188   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.

42.0.189   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.

42.0.190   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.

42.0.191   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.

42.0.192   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.

42.0.193 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)

42.0.194   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.

42.0.195 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.

42.0.196 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

42.0.197 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
```

42.0.198   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.

42.0.199   How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:
Example:

window1.ModifiedMBS=true

42.0.200   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.

42.0.201 How to show Keyboard Viewer programmatically?

Platform: macOS.

Answer: Use Xojo or AppleScript to launch the KeyboardViewerServer.app.
Example:

dim a as new AppleScriptMBS
dim text as string
dim lines(-1) as string

lines.append "set theApplication to ""KeyboardViewerServer"""
lines.append "set thePath to ""/System/Library/Components/KeyboardViewer.component/Contents/Shared-Support/KeyboardViewerServer.app"""
lines.append ""
lines.append "set POSIXPath to ((POSIX file thePath) as string)"
lines.append "tell application ""System Events"" to set isRunning to 0 <(count (application processes whose name is theApplication))"
lines.append "if isRunning then tell application POSIXPath to quit"
lines.append "delay 0.15"
lines.append ""
lines.append "ignoring application responses"
lines.append " tell application POSIXPath to run"
lines.append "end ignoring"

text=join(lines,EndOfLine.macintosh)

a.Compile text
a.Execute

Notes: AppleScript code:

set theApplication to "KeyboardViewerServer"
set thePath to "/System/Library/Components/KeyboardViewer.component/Contents/SharedSupport/KeyboardViewerServer.app"

set POSIXPath to ((POSIX file thePath) as string)
tell application "System Events" to set isRunning to 0 <(count (application processes whose name is theApplication))
if isRunning then tell application POSIXPath to quit
delay 0.15

ignoring application responses
tell application POSIXPath to run
end ignoring

42.0.202 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.

42.0.203 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

42.0.204   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

42.0.205   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.

42.0.206   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.

42.0.207   How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.
Example:

msgbox "This String contains ""quotes""."

42.0.208   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

42.0.209   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!

42.0.210   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.

42.0.211 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".

42.0.212 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.

42.0.213   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.

42.0.214   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.

42.0.215   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.

42.0.216 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.

42.0.217 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.

42.0.218 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)

42.0.219   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.

42.0.220   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

42.0.221 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.

42.0.222 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
```

42.0.223   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

42.0.224   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)

42.0.225   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?

42.0.226 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.

42.0.227 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.

42.0.228   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.

42.0.229 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.

42.0.230 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.

42.0.231 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.

42.0.232   What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

42.0.233   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
```

42.0.234   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.

42.0.235   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.

42.0.236   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.

42.0.237   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.

42.0.238   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.TOOL-BAR, 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, COR-REL, COS, COSH, COUNT, COUNTA, COUNTBLANK, COUNTIF, COVAR, CREATE.OBJECT, CRIT-BINOM, CUSTOM.REPEAT, CUSTOM.UNDO, DATE, DATEDIF, DATESTRING, DATEVALUE, DAV-ERAGE, DAY, DAYS360, DB, DBCS, DCOUNT, DCOUNTA, DDB, DEGREES, DELETE.BAR, DELETE.COM-MAND, 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, EVAL-UATE, 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.TOOL-BAR, GET.WINDOW, GET.WORKBOOK, GET.WORKSPACE, GETPIVOTDATA, GOTO, GROUP, GROWTH, HALT, HARMEAN, HELP, HLOOKUP, HOUR, HYPERLINK, HYPGEOMDIST, IF, IN-DEX, INDIRECT, INFO, INITIATE, INPUT, INT, INTERCEPT, IPMT, IRR, ISBLANK, ISERR, ISER-ROR, 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.COM-MAND, N, NA, NAMES, NEGBINOMDIST, NEXT, NORMDIST, NORMINV, NORMSDIST, NORM-SINV, NOT, NOTE, NOW, NPER, NPV, NUMBERSTRING, ODD, OFFSET, OPEN.DIALOG, OP-TIONS.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, ROUND-BAHTUP, 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, STANDARD-IZE, STDEV, STDEVA, STDEVP, STDEVPA, STEP, STEYX, SUBSTITUTE, SUBTOTAL, SUM, SUMIF, SUMPRODUCT, SUMSQ, SUMX2MY2, SUMX2PY2, SUMXMY2, SYD, T, TAN, TANH, TDIST, TER-MINATE, TEXT, TEXT.BOX, TEXTREF, THAIDAYOFWEEK, THAIDIGIT, THAIMONTHOFYEAR, THAINUMSOUND, THAINUMSTRING, THAISTRINGLENGTH, THAIYEAR, TIME, TIMEVALUE, TINV, TODAY, TRANSPOSE, TREND, TRIM, TRIMMEAN, TRUE, TRUNC, TTEST, TYPE, UNREG-ISTER, UPPER, USDOLLAR, USERDEFINED, VALUE, VAR, VARA, VARP, VARPA, VDB, VIEW.GET, VLOOKUP, VOLATILE, WEEKDAY, WEIBULL, WHILE, WINDOW.TITLE, WINDOWS, YEAR and ZTEST.

42.0.239   What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

42.0.240   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.

42.0.241   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.

42.0.242   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.

42.0.243   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.

42.0.244   Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

42.0.245   Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

42.0.246   Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

42.0.247   Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

42.0.248   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.

42.0.249   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.

42.0.250   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.

42.0.251   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, XLSMerged-CellsMBS, XLSRowMBS and XLSSheetMBS.

42.0.252   Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:
http://www.wotsit.org

42.0.253   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

42.0.254   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.

42.0.255   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.

42.0.256   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.

42.0.257   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.

42.0.258   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.

42.0.259   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.

42.0.260   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.

42.0.261   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.

42.0.262   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.

42.0.263   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.

42.0.264   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

42.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

42.0.266   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]

textasciitilde ' for no thousand separator. The default is '
textasciitilde ', which can be modified using BaseChart.setNumberFormat.

[c]

[d]

textasciitilde ' for no negative sign character. The default is '-', which can be modified using BaseChart.setNumberFormat.

The thousand separator. Should be a non-alphanumeric character (not 0-9, A-Z, a-z). Use '

The decimal point character. The default is '.', which can be modified using BaseChart.setNumberFormat.
The negative sign character. Use '

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,,