

MBS MacBase 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 MacBase Plugin

0.2 Content

- 1 List of all topics 3
- 2 List of all classes 117
- 3 All items in this plugin 121
- 17 List of Questions in the FAQ 1133
- 18 The FAQ 1143

Chapter 1

List of Topics

- 5 Cocoa Controls 549
 - 5.1.1 class ContainerControl 549
 - * 5.1.3 NSViewMBS as NSViewMBS 549

• 5 Cocoa Controls	549
– 11.1.1 class Control	1045
* 11.1.3 NSViewMBS as NSViewMBS	1045
– 5.2.1 class DesktopContainer	550
* 5.2.3 NSViewMBS as NSViewMBS	550

	5
• 5 Cocoa Controls	549
– 11.2.1 class DesktopControl	1046
* 11.2.3 NSViewMBS as NSViewMBS	1046

- **16 Window** 1129
 - 16.1.1 class DesktopWindow 1129
 - * 16.1.4 NSWindowMBS as NSWindowMBS 1130

	7
• 4 Cocoa	147
– 16.1.1 class DesktopWindow	1129
* 16.1.3 NSPanelMBS as NSPanelMBS	1129

- **16 Window** 1129
 - 16.1.1 class DesktopWindow 1129
 - * 16.1.4 NSWindowMBS as NSWindowMBS 1130

- **13 DiscRecording** 1095
 - 13.1.1 class DRNotificationCenterMBS 1095
 - * 13.1.3 addObserver(observer as NSNotificationCenterObserverMBS, name as string="", theObject as Variant=nil) 1095
 - * 13.1.4 Constructor 1096
 - * 13.1.5 removeObserver(observer as NSNotificationCenterObserverMBS, name as string, theObject as Variant=nil) 1096

• 9 Cocoa Text	925
– 9.1.1 class NSAttributedStringMBS	925
* 9.1.3 AsCFAttributedString as Variant	926
* 9.1.4 attributeAtIndex(name as string, location as UInt64) as Variant	926
* 9.1.5 attributeAtIndex(name as string, location as UInt64, inRange as NSRangeMBS) as Variant	927
* 9.1.6 attributeAtIndex2(name as string, location as UInt64, byref effectiveRange as NSRangeMBS) as Variant	928
* 9.1.7 attributeAtIndex2(name as string, location as UInt64, byref longestEffectiveRange as NSRangeMBS, inRange as NSRangeMBS) as Variant	929
* 9.1.8 attributedStringWithAttachment(attachment as NSTextAttachmentMBS) as NSAttributedStringMBS	929
* 9.1.9 attributedStringWithAttributedString(text as NSAttributedStringMBS) as NSAttributedStringMBS	930
* 9.1.10 attributedStringWithDocFormat(data as memoryblock) as NSAttributedStringMBS	930
* 9.1.11 attributedStringWithDocFormat(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	930
* 9.1.12 attributedStringWithHTML(data as memoryblock) as NSAttributedStringMBS	931
* 9.1.13 attributedStringWithHTML(data as memoryblock, BaseURL as string) as NSAttributedStringMBS	931
* 9.1.14 attributedStringWithHTML(data as memoryblock, BaseURL as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	931
* 9.1.15 attributedStringWithHTML(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	932
* 9.1.16 attributedStringWithHTMLOld(data as string) as NSAttributedStringMBS	932
* 9.1.17 attributedStringWithPath(file as folderitem) as NSAttributedStringMBS	932
* 9.1.18 attributedStringWithPath(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	933
* 9.1.19 attributedStringWithPath(path as string) as NSAttributedStringMBS	933
* 9.1.20 attributedStringWithPath(path as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	933
* 9.1.21 attributedStringWithRTF(data as memoryblock) as NSAttributedStringMBS	934
* 9.1.22 attributedStringWithRTF(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	934
* 9.1.23 attributedStringWithRTFD(data as memoryblock) as NSAttributedStringMBS	934
* 9.1.24 attributedStringWithRTFD(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	934
* 9.1.25 attributedStringWithString(text as string) as NSAttributedStringMBS	935
* 9.1.26 attributedStringWithString(text as string, withAttributes as dictionary) as NSAttributedStringMBS	935
* 9.1.27 attributedStringWithURL(file as folderitem) as NSAttributedStringMBS	935
* 9.1.28 attributedStringWithURL(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS	936

- * 9.1.29 attributedStringWithURL(url as string) as NSAttributedStringMBS 936
- * 9.1.30 attributedStringWithURL(url as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936
- * 9.1.31 attributedSubstringFromRange(range as NSRangeMBS) as NSAttributedStringMBS 937
- * 9.1.32 attributesAtIndex(location as UInt64) as dictionary 937
- * 9.1.33 attributesAtIndex(location as UInt64, inRange as NSRangeMBS) as dictionary 937
- * 9.1.34 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS) as dictionary 938
- * 9.1.35 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS, inRange as NSRangeMBS) as dictionary 938
- * 9.1.36 Constructor 939
- * 9.1.37 Convert_Operator as string 939
- * 9.1.38 copy as NSAttributedStringMBS 939
- * 9.1.39 CopyToClipboard as Boolean 939
- * 9.1.40 dataFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil, byref error as NSErrorMBS) as memoryblock 940
- * 9.1.41 docFormatFromRange(documentAttributes as dictionary = nil) as MemoryBlock 940
- * 9.1.42 docFormatFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock 940
- * 9.1.43 fileWrapperFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil, byref Error as NSErrorMBS) as NSFileWrapperMBS 941
- * 9.1.44 FromClipboard as NSAttributedStringMBS 942
- * 9.1.45 GeneratePDF(PrintOptions as Variant = nil) as MemoryBlock 942
- * 9.1.46 htmlString as string 942
- * 9.1.47 initWithAttributedString(text as NSAttributedStringMBS) as boolean 943
- * 9.1.48 initWithDocFormat(data as MemoryBlock) as boolean 943
- * 9.1.49 initWithDocFormat(data as memoryblock, byref documentAttributes as dictionary) as boolean 943
- * 9.1.50 initWithHTML(data as MemoryBlock) as boolean 943
- * 9.1.51 initWithHTML(data as MemoryBlock, BaseURL as string) as boolean 944
- * 9.1.52 initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean 945
- * 9.1.53 initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean 945
- * 9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean 946
- * 9.1.55 initWithHTMLOld(data as string) as boolean 947
- * 9.1.56 initWithPath(file as folderitem) as boolean 947
- * 9.1.57 initWithPath(file as folderitem, byref documentAttributes as dictionary) as boolean 947
- * 9.1.58 initWithPath(path as string) as boolean 948
- * 9.1.59 initWithPath(path as string, byref documentAttributes as dictionary) as boolean 948

* 9.1.60 initWithRTF(data as MemoryBlock) as boolean	948
* 9.1.61 initWithRTF(data as memoryblock, byref documentAttributes as dictionary) as boolean	949
* 9.1.62 initWithRTFD(data as MemoryBlock) as boolean	949
* 9.1.63 initWithRTFD(data as memoryblock, byref documentAttributes as dictionary) as boolean	949
* 9.1.64 initWithString(text as string) as boolean	949
* 9.1.65 initWithString(text as string, withAttributes as Dictionary) as boolean	950
* 9.1.66 initWithURL(file as folderitem) as boolean	950
* 9.1.67 initWithURL(file as folderitem, byref documentAttributes as dictionary) as boolean	951
* 9.1.68 initWithURL(url as string) as boolean	952
* 9.1.69 initWithURL(url as string, byref documentAttributes as dictionary) as boolean	952
* 9.1.70 isEqualToAttributedString(other as NSAttributedStringMBS) as Boolean	952
* 9.1.71 itemNumberInTextList(list as NSTextListMBS, location as Integer) as Integer	953
* 9.1.72 lineRangeForRange(range as NSRangeMBS) as NSRangeMBS	953
* 9.1.73 mutableCopy as NSMutableAttributedStringMBS	954
* 9.1.74 NSAttachmentAttributeName as string	954
* 9.1.75 NSAuthorDocumentAttribute as string	954
* 9.1.76 NSBackgroundColorAttributeName as string	955
* 9.1.77 NSBackgroundColorDocumentAttribute as string	956
* 9.1.78 NSBaselineOffsetAttributeName as string	956
* 9.1.79 NSBaseURLDocumentOption as string	956
* 9.1.80 NSBottomMarginDocumentAttribute as string	956
* 9.1.81 NSCategoryDocumentAttribute as string	957
* 9.1.82 NSCharacterEncodingDocumentAttribute as string	957
* 9.1.83 NSCharacterEncodingDocumentOption as string	957
* 9.1.84 NSCharacterShapeAttributeName as string	958
* 9.1.85 NSCocoaVersionDocumentAttribute as string	958
* 9.1.86 NSCommentDocumentAttribute as string	958
* 9.1.87 NSCompanyDocumentAttribute as string	958
* 9.1.88 NSConvertedDocumentAttribute as string	959
* 9.1.89 NSCopyrightDocumentAttribute as string	959
* 9.1.90 NSCreationTimeDocumentAttribute as string	959
* 9.1.91 NSCursorAttributeName as string	959
* 9.1.92 NSDefaultAttributesDocumentAttribute as string	959
* 9.1.93 NSDefaultAttributesDocumentOption as string	960
* 9.1.94 NSDefaultTabIntervalDocumentAttribute as string	960
* 9.1.95 NSDocFormatTextDocumentType as string	960
* 9.1.96 NSDocumentTypeDocumentAttribute as string	960
* 9.1.97 NSDocumentTypeDocumentOption as string	961
* 9.1.98 NSEditorDocumentAttribute as string	961

* 9.1.99	NSExcludedElementsDocumentAttribute as string	961
* 9.1.100	NSExpansionAttributeName as string	961
* 9.1.101	NSFileTypeDocumentAttribute as string	962
* 9.1.102	NSFileTypeDocumentOption as string	962
* 9.1.103	NSFontAttributeName as string	963
* 9.1.104	NSForegroundColorAttributeName as string	963
* 9.1.105	NSGlyphInfoAttributeName as string	964
* 9.1.106	NSHTMLTextDocumentType as string	964
* 9.1.107	NSHyphenationFactorDocumentAttribute as string	964
* 9.1.108	NSKernAttributeName as string	964
* 9.1.109	NSKeywordsDocumentAttribute as string	965
* 9.1.110	NSLeftMarginDocumentAttribute as string	965
* 9.1.111	NSLigatureAttributeName as string	965
* 9.1.112	NSLinkAttributeName as string	965
* 9.1.113	NSMacSimpleTextDocumentType as string	965
* 9.1.114	NSManagerDocumentAttribute as string	966
* 9.1.115	NSMarkedClauseSegmentAttributeName as string	966
* 9.1.116	NSModificationTimeDocumentAttribute as string	966
* 9.1.117	NSObliquenessAttributeName as string	966
* 9.1.118	NSOfficeOpenXMLTextDocumentType as string	966
* 9.1.119	NSOpenDocumentTextDocumentType as string	967
* 9.1.120	NSPaperSizeDocumentAttribute as string	967
* 9.1.121	NSParagraphStyleAttributeName as string	967
* 9.1.122	NSPlainTextDocumentType as string	968
* 9.1.123	NSPrefixSpacesDocumentAttribute as string	968
* 9.1.124	NSReadOnlyDocumentAttribute as string	968
* 9.1.125	NSRightMarginDocumentAttribute as string	968
* 9.1.126	NSRTFDTextDocumentType as string	969
* 9.1.127	NSRTFTextDocumentType as string	969
* 9.1.128	NSShadowAttributeName as string	969
* 9.1.129	NSSpellingStateAttributeName as string	969
* 9.1.130	NSStrikethroughColorAttributeName as string	969
* 9.1.131	NSStrikethroughStyleAttributeName as string	970
* 9.1.132	NSStrokeColorAttributeName as string	970
* 9.1.133	NSStrokeWidthAttributeName as string	970
* 9.1.134	NSSubjectDocumentAttribute as string	971
* 9.1.135	NSSuperscriptAttributeName as string	971
* 9.1.136	NSTextAlternativesAttributeName as string	971
* 9.1.137	NSTextEffectAttributeName as string	972
* 9.1.138	NSTextEffectLetterpressStyle as string	972
* 9.1.139	NSTextEncodingNameDocumentAttribute as string	972
* 9.1.140	NSTextEncodingNameDocumentOption as string	972

* 9.1.141 NSTextLayoutSectionOrientation as string	972
* 9.1.142 NSTextLayoutSectionRange as string	973
* 9.1.143 NSTextLayoutSectionsAttribute as string	973
* 9.1.144 NSTextSizeMultiplierDocumentOption as string	973
* 9.1.145 NSTimeoutDocumentOption as string	973
* 9.1.146 NSTitleDocumentAttribute as string	974
* 9.1.147 NSToolTipAttributeName as string	974
* 9.1.148 NSTopMarginDocumentAttribute as string	974
* 9.1.149 NSUnderlineColorAttributeName as string	974
* 9.1.150 NSUnderlineStyleAttributeName as string	975
* 9.1.151 NSVerticalGlyphFormAttributeName as string	975
* 9.1.152 NSViewModeDocumentAttribute as string	975
* 9.1.153 NSViewSizeDocumentAttribute as string	975
* 9.1.154 NSViewZoomDocumentAttribute as string	975
* 9.1.155 NSWebArchiveTextDocumentType as string	976
* 9.1.156 NSWebPreferencesDocumentOption as string	976
* 9.1.157 NSWebResourceLoadDelegateDocumentOption as string	976
* 9.1.158 NSWordMLTextDocumentType as string	976
* 9.1.159 NSWritingDirectionAttributeName as string	977
* 9.1.160 paragraphRangeForRange(range as NSRangeMBS) as NSRangeMBS	977
* 9.1.161 rangeOfTextBlock(textBlock as NSTextBlockMBS, location as Integer) as NSRangeMBS	978
* 9.1.162 rangeOfTextList(list as NSTextListMBS, location as Integer) as NSRangeMBS	978
* 9.1.163 rangeOfTextTable(textTable as NSTextTableMBS, location as Integer) as NSRangeMBS	978
* 9.1.164 rtf as MemoryBlock	978
* 9.1.165 RTFDFileWrapperFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as NSFileWrapperMBS	979
* 9.1.166 RTFDFromRange(documentAttributes as dictionary = nil) as MemoryBlock	979
* 9.1.167 RTFDFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock	980
* 9.1.168 RTFFFromRange(documentAttributes as dictionary = nil) as MemoryBlock	980
* 9.1.169 RTFFFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock	980
* 9.1.171 containsAttachments as boolean	981
* 9.1.172 Handle as Integer	981
* 9.1.173 length as Integer	982
* 9.1.174 Range as NSRangeMBS	982
* 9.1.175 text as string	982

	15
• 6 Cocoa Drawing	551
– 6.1.1 class NSBezierPathMBS	551
* 6.1.3 appendBezierPath(path as NSBezierPathMBS)	552
* 6.1.4 appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double)	552
* 6.1.5 appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean)	552
* 6.1.6 appendBezierPathWithArc(point1 as NSPointMBS, point2 as NSPointMBS, radius as Double)	553
* 6.1.7 appendBezierPathWithGlyph(glyph as Integer, font as NSFontMBS)	554
* 6.1.8 appendBezierPathWithGlyphs(glyphs() as Integer, font as NSFontMBS)	554
* 6.1.9 appendBezierPathWithOvalInRect(rect as NSRectMBS)	555
* 6.1.10 appendBezierPathWithPoints(points() as NSPointMBS)	555
* 6.1.11 appendBezierPathWithRect(rect as NSRectMBS)	555
* 6.1.12 appendBezierPathWithRoundedRect(rect as NSRectMBS, xRadius as Double, yRadius as Double)	556
* 6.1.13 bezierPath as NSBezierPathMBS	556
* 6.1.14 bezierPathByFlatteningPath as NSBezierPathMBS	556
* 6.1.15 bezierPathByReversingPath as NSBezierPathMBS	556
* 6.1.16 bezierPathWithOvalInRect(r as NSRectMBS) as NSBezierPathMBS	557
* 6.1.17 bezierPathWithRect(r as NSRectMBS) as NSBezierPathMBS	557
* 6.1.18 bezierPathWithRoundedRect(r as NSRectMBS, xRadius as Double, yRadius as Double) as NSBezierPathMBS	557
* 6.1.19 closePath	558
* 6.1.20 Constructor	558
* 6.1.21 containsPoint(p as NSPointMBS) as boolean	558
* 6.1.22 copy as NSBezierPathMBS	559
* 6.1.23 curveToPoint(endPoint as NSPointMBS, controlPoint1 as NSPointMBS, controlPoint2 as NSPointMBS)	559
* 6.1.24 elementAtIndex(index as Integer) as Integer	559
* 6.1.25 elementAtIndex(index as Integer, byref associatedPoints() as NSPointMBS) as Integer	560
* 6.1.26 elementCount as Integer	560
* 6.1.27 getLineDash(byref pattern() as Double, byref count as Integer, byref phase as Double)	560
* 6.1.28 isEmpty as boolean	561
* 6.1.29 lineToPoint(p as NSPointMBS)	561
* 6.1.30 moveToPoint(p as NSPointMBS)	561
* 6.1.31 relativeCurveToPoint(endPoint as NSPointMBS, controlPoint1 as NSPointMBS, controlPoint2 as NSPointMBS)	562
* 6.1.32 relativeLineToPoint(p as NSPointMBS)	562
* 6.1.33 relativeMoveToPoint(p as NSPointMBS)	563

* 6.1.34	removeAllPoints	563
* 6.1.35	setAssociatedPoints(points() as NSPointMBS, index as Integer)	563
* 6.1.36	setLineDash(pattern() as Double, phase as Double)	564
* 6.1.37	transformUsingAffineTransform(transform as Variant)	565
* 6.1.39	Bounds as NSRectMBS	565
* 6.1.40	ControlPointBounds as NSRectMBS	566
* 6.1.41	CurrentPoint as NSPointMBS	566
* 6.1.42	defaultFlatness as Double	566
* 6.1.43	defaultLineCapStyle as Integer	567
* 6.1.44	defaultLineJoinStyle as Integer	567
* 6.1.45	defaultLineWidth as Double	567
* 6.1.46	defaultMiterLimit as Double	567
* 6.1.47	defaultWindingRule as Integer	568
* 6.1.48	Handle as Integer	568
* 6.1.49	flatness as Double	568
* 6.1.50	lineCapStyle as Integer	568
* 6.1.51	lineJoinStyle as Integer	569
* 6.1.52	lineWidth as Double	569
* 6.1.53	miterLimit as Double	570
* 6.1.54	windingRule as Integer	570
– 6.2.1	class NSBitmapImageRepMBS	572
* 6.2.3	bitmapImageRepByConvertingToColorSpace(colorSpace as NSColorSpaceMBS, renderingIntent as Integer) as NSBitmapImageRepMBS	572
* 6.2.4	bitmapImageRepByRetaggingWithColorSpace(newSpace as NSColorSpaceMBS) as NSBitmapImageRepMBS	573
* 6.2.5	BMPRepresentation(properties as dictionary = nil) as Memoryblock	573
* 6.2.6	canBeCompressedUsing(compression as Integer) as Boolean	573
* 6.2.7	Constructor(data as Memoryblock)	574
* 6.2.8	Constructor(pic as Picture)	574
* 6.2.9	Constructor(pixelsWide as Integer, pixelsHigh as Integer, bitsPerSample as Integer, samplesPerPixel as Integer, hasAlpha as boolean, colorSpaceName as string, bytesPerRow as Integer, bitsPerPixel as Integer)	575
* 6.2.10	GIFRepresentation(properties as dictionary = nil) as Memoryblock	576
* 6.2.11	imageRepWithCGImage(CGImage as Variant) as NSBitmapImageRepMBS	576
* 6.2.12	imageRepWithCGImageRef(CGImageHandle as Integer) as NSBitmapImageRepMBS	577
* 6.2.13	imageRepWithCIImage(CIImage as variant) as NSBitmapImageRepMBS	577
* 6.2.14	imageRepWithCIImageRef(CIImageHandle as Integer) as NSBitmapImageRepMBS	578
* 6.2.15	imageRepWithData(data as Memoryblock) as NSBitmapImageRepMBS	578
* 6.2.16	JPEGRepresentation(properties as dictionary = nil) as Memoryblock	579
* 6.2.17	NSImageColorSyncProfileData as string	579

* 6.2.18	NSImageCompressionFactor as string	579
* 6.2.19	NSImageCompressionMethod as string	579
* 6.2.20	NSImageCurrentFrame as string	579
* 6.2.21	NSImageCurrentFrameDuration as string	580
* 6.2.22	NSImageDitherTransparency as string	580
* 6.2.23	NSImageEXIFData as string	580
* 6.2.24	NSImageFallbackBackgroundColor as string	580
* 6.2.25	NSImageFrameCount as string	581
* 6.2.26	NSImageGamma as string	581
* 6.2.27	NSImageInterlaced as string	581
* 6.2.28	NSImageLoopCount as string	581
* 6.2.29	NSImageProgressive as string	581
* 6.2.30	NSImageRGBColorTable as string	582
* 6.2.31	PNGRepresentation(properties as dictionary = nil) as Memoryblock	582
* 6.2.32	TIFFRepresentation as Memoryblock	582
* 6.2.33	TIFFRepresentation(properties as dictionary = nil) as Memoryblock	582
* 6.2.35	bitmapData as Ptr	583
* 6.2.36	bitmapFormat as Integer	583
* 6.2.37	bitsPerPixel as Integer	583
* 6.2.38	bytesPerPlane as Integer	583
* 6.2.39	bytesPerRow as Integer	583
* 6.2.40	CGImage as Variant	584
* 6.2.41	colorSpace as NSColorSpaceMBS	584
* 6.2.42	isPlanar as Boolean	584
* 6.2.43	numberOfPlanes as Integer	584
* 6.2.44	samplesPerPixel as Integer	585
* 6.2.45	valueForProperty(key as string) as Variant	585

• 4 Cocoa	147
– 4.1.1 class NSBundleMBS	147
* 4.1.3 allBundles as NSBundleMBS()	148
* 4.1.4 allFrameworks as NSBundleMBS()	148
* 4.1.5 builtInPlugInsFolder as folderitem	149
* 4.1.6 builtInPlugInsPath as string	149
* 4.1.7 bundleFolder as folderitem	149
* 4.1.8 bundleIdentifier as string	149
* 4.1.9 bundlePath as string	150
* 4.1.10 bundleWithIdentifier(identifier as string) as NSBundleMBS	150
* 4.1.11 bundleWithPath(path as folderitem) as NSBundleMBS	150
* 4.1.12 bundleWithPath(path as string) as NSBundleMBS	151
* 4.1.13 Constructor(path as folderitem)	151
* 4.1.14 Constructor(path as string)	152
* 4.1.15 developmentLocalization as string	152
* 4.1.16 executableArchitectures as Integer()	153
* 4.1.17 executableFile as folderitem	154
* 4.1.18 executablePath as string	154
* 4.1.19 infoDictionary as dictionary	154
* 4.1.20 isLoading as boolean	155
* 4.1.21 load as boolean	155
* 4.1.22 localizations as string()	155
* 4.1.23 localizedInfoDictionary as dictionary	156
* 4.1.24 localizedStringForKey(key as string, value as string="", tableName as string="") as string	156
* 4.1.25 mainBundle as NSBundleMBS	157
* 4.1.26 pathForResource(name as string) as folderitem	157
* 4.1.27 pathForResource(name as string, extension as string) as folderitem	158
* 4.1.28 pathForResource(name as string, extension as string, subpath as string) as folderitem	158
* 4.1.29 pathForResource(name as string, extension as string, subpath as string, localization-Name as string) as folderitem	159
* 4.1.30 pathForResource(name as string) as folderitem	160
* 4.1.31 preferredLocalizations as string()	160
* 4.1.32 privateFrameworksFolder as folderitem	160
* 4.1.33 privateFrameworksPath as string	160
* 4.1.34 resourceFolder as folderitem	160
* 4.1.35 resourcePath as string	161
* 4.1.36 sharedFrameworksFolder as folderitem	161
* 4.1.37 sharedFrameworksPath as string	161
* 4.1.38 sharedSupportFolder as folderitem	161
* 4.1.39 sharedSupportPath as string	161
* 4.1.40 unload as boolean	162
* 4.1.42 Handle as Integer	162

	19
• 8 Cocoa Networking	797
– 8.1.1 class NSCachedURLResponseMBS	797
* 8.1.3 Constructor(response as NSURLResponseMBS, data as MemoryBlock, userInfo as Dictionary = nil, storagePolicy as Integer = 0)	797
* 8.1.4 copy as NSCachedURLResponseMBS	798
* 8.1.6 Data as MemoryBlock	798
* 8.1.7 Handle as Integer	798
* 8.1.8 Response as NSURLResponseMBS	798
* 8.1.9 StoragePolicy as Integer	799
* 8.1.10 UserInfo as Dictionary	799

• 4 Cocoa	147
– 4.2.1 class NSCalendarMBS	164
* 4.2.3 AMSymbol as string	164
* 4.2.4 autoupdatingCurrentCalendar as NSCalendarMBS	164
* 4.2.5 calendarIdentifier as string	165
* 4.2.6 calendarWithIdentifier(identifier as String) as NSCalendarMBS	165
* 4.2.7 componentsInTimeZone(timezone as NSTimeZoneMBS, date as Date) as NSDateComponentsMBS	165
* 4.2.8 componentsInTimeZone(timezone as NSTimeZoneMBS, date as DateTime) as NSDateComponentsMBS	165
* 4.2.9 Constructor	166
* 4.2.10 Constructor(identifier as string)	166
* 4.2.11 copy as NSCalendarMBS	166
* 4.2.12 currentCalendar as NSCalendarMBS	166
* 4.2.13 dateByAddingComponents(components as NSDateComponentsMBS, toDate as Date, Options as Integer = 0) as Date	167
* 4.2.14 dateFromComponents(components as NSDateComponentsMBS) as Date	167
* 4.2.15 dateTimeByAddingComponents(components as NSDateComponentsMBS, toDate as DateTime, Options as Integer = 0) as DateTime	167
* 4.2.16 dateTimeFromComponents(components as NSDateComponentsMBS) as DateTime	168
* 4.2.17 description as string	168
* 4.2.18 NSCalendarIdentifierBuddhist as String	168
* 4.2.19 NSCalendarIdentifierChinese as String	168
* 4.2.20 NSCalendarIdentifierCoptic as String	168
* 4.2.21 NSCalendarIdentifierEthiopicAmeteAlem as String	169
* 4.2.22 NSCalendarIdentifierEthiopicAmeteMihret as String	169
* 4.2.23 NSCalendarIdentifierGregorian as String	169
* 4.2.24 NSCalendarIdentifierHebrew as String	169
* 4.2.25 NSCalendarIdentifierIndian as String	169
* 4.2.26 NSCalendarIdentifierIslamic as String	169
* 4.2.27 NSCalendarIdentifierIslamicCivil as String	170
* 4.2.28 NSCalendarIdentifierIslamicTabular as String	170
* 4.2.29 NSCalendarIdentifierIslamicUmmAlQura as String	170
* 4.2.30 NSCalendarIdentifierISO8601 as String	170
* 4.2.31 NSCalendarIdentifierJapanese as String	170
* 4.2.32 NSCalendarIdentifierPersian as String	170
* 4.2.33 NSCalendarIdentifierRepublicOfChina as String	171
* 4.2.34 PMSymbol as string	171
* 4.2.35 Print	171
* 4.2.37 Handle as Integer	171
* 4.2.38 firstWeekday as Integer	171
* 4.2.39 locale as NSLocaleMBS	171

	21
* 4.2.40 minimumDaysInFirstWeek as Integer	172
* 4.2.41 timeZone as NSTimeZoneMBS	172
– 4.3.1 class NSCharacterSetMBS	173
* 4.3.3 alphanumericCharacterSet as NSCharacterSetMBS	174
* 4.3.4 bitmapRepresentation as MemoryBlock	174
* 4.3.5 capitalizedLetterCharacterSet as NSCharacterSetMBS	174
* 4.3.6 characterIsMember(Character as Integer) as boolean	175
* 4.3.7 characterSetWithBitmapRepresentation(data as MemoryBlock) as NSCharacterSetMBS	175
* 4.3.8 characterSetWithCharactersInString(aString as string) as NSCharacterSetMBS	175
* 4.3.9 characterSetWithContentsOfFile(aString as string) as NSCharacterSetMBS	176
* 4.3.10 characterSetWithContentsOfFile(file as folderitem) as NSCharacterSetMBS	176
* 4.3.11 characterSetWithRange(r as NSRangeMBS) as NSCharacterSetMBS	176
* 4.3.12 componentsSeparatedByCharactersInSet(s as string) as String()	177
* 4.3.13 Constructor	177
* 4.3.14 controlCharacterSet as NSCharacterSetMBS	177
* 4.3.15 copy as NSCharacterSetMBS	178
* 4.3.16 decimalDigitCharacterSet as NSCharacterSetMBS	178
* 4.3.17 decomposableCharacterSet as NSCharacterSetMBS	178
* 4.3.18 hasMemberInPlane(thePlane as Integer) as boolean	179
* 4.3.19 illegalCharacterSet as NSCharacterSetMBS	179
* 4.3.20 invertedSet as NSCharacterSetMBS	179
* 4.3.21 isSupersetOfSet(theOtherSet as NSCharacterSetMBS) as boolean	179
* 4.3.22 letterCharacterSet as NSCharacterSetMBS	179
* 4.3.23 longCharacterIsMember(theLongChar as Integer) as boolean	180
* 4.3.24 lowercaseLetterCharacterSet as NSCharacterSetMBS	180
* 4.3.25 mutableCopy as NSMutableCharacterSetMBS	180
* 4.3.26 newlineCharacterSet as NSCharacterSetMBS	180
* 4.3.27 nonBaseCharacterSet as NSCharacterSetMBS	181
* 4.3.28 Operator_Convert as string	181
* 4.3.29 punctuationCharacterSet as NSCharacterSetMBS	181
* 4.3.30 rangeOfCharacterFromSet(s as string, options as Integer = 0, searchRange as NSRangeMBS = nil) as NSRangeMBS	182
* 4.3.31 stringByTrimmingCharactersInSet(s as string) as String	182
* 4.3.32 symbolCharacterSet as NSCharacterSetMBS	183
* 4.3.33 uppercaseLetterCharacterSet as NSCharacterSetMBS	183
* 4.3.34 whitespaceAndNewlineCharacterSet as NSCharacterSetMBS	183
* 4.3.35 whitespaceCharacterSet as NSCharacterSetMBS	184
* 4.3.37 CharacterCount as Integer	184
* 4.3.38 Handle as Integer	184
* 4.3.39 StringValue as string	184
– 4.4.1 class NSCoderMBS	186

* 4.4.3 allowsKeyedCoding as boolean	187
* 4.4.4 Constructor	187
* 4.4.5 containsValueForKey(key as string) as boolean	187
* 4.4.6 decodeBool(key as string) as boolean	187
* 4.4.7 decodeBytes(key as string) as MemoryBlock	188
* 4.4.8 decodeCFObjectMBS(key as string) as Variant	188
* 4.4.9 decodeDictionary(key as string) as Dictionary	188
* 4.4.10 decodeDouble(key as string) as Double	188
* 4.4.11 decodeFloat(key as string) as single	188
* 4.4.12 decodeInt32(key as string) as Int32	188
* 4.4.13 decodeInt64(key as string) as Int64	189
* 4.4.14 decodeNSURLFile(key as string) as folderitem	189
* 4.4.15 decodeNSURLString(key as string) as String	189
* 4.4.16 decodePoint(key as string) as NSPointMBS	189
* 4.4.17 decodeRect(key as string) as NSRectMBS	189
* 4.4.18 decodeSize(key as string) as NSSizeMBS	189
* 4.4.19 decodeString(key as string) as string	190
* 4.4.20 decodeTypedStream(Data as MemoryBlock) as variant	190
* 4.4.21 encodeBool(value as boolean, key as string)	190
* 4.4.22 encodeBytes(value as MemoryBlock, key as string)	190
* 4.4.23 encodeCFObjectMBS(value as Variant, key as string)	190
* 4.4.24 encodeDictionary(value as Dictionary, key as string)	191
* 4.4.25 encodeDouble(value as Double, key as string)	191
* 4.4.26 encodeFloat(value as single, key as string)	191
* 4.4.27 encodeInt32(value as Int32, key as string)	191
* 4.4.28 encodeInt64(value as Int64, key as string)	191
* 4.4.29 encodeNSURLFile(value as folderitem, key as string)	191
* 4.4.30 encodeNSURLString(value as String, key as string)	192
* 4.4.31 encodePoint(value as NSPointMBS, key as string)	192
* 4.4.32 encodeRect(value as NSRectMBS, key as string)	192
* 4.4.33 encodeSize(value as NSSizeMBS, key as string)	192
* 4.4.34 encodeString(value as string, key as string)	192
* 4.4.35 systemVersion as Integer	192
* 4.4.37 Handle as Integer	193

• 6 Cocoa Drawing	551
– 6.3.1 class NSColorListMBS	587
* 6.3.3 colorWithKey(key as string) as NSColorMBS	587
* 6.3.4 Create(name as string) as boolean	587
* 6.3.5 Create(name as string, path as string) as boolean	587
* 6.3.6 insertColor(theColor as NSColorMBS, key as string, index as Integer)	587
* 6.3.7 isEditable as Boolean	588
* 6.3.8 Load(name as string) as boolean	588
* 6.3.9 name as string	588
* 6.3.10 removeColorWithKey(key as string)	588
* 6.3.11 removeFile	588
* 6.3.12 setColor(theColor as NSColorMBS, key as string)	589
* 6.3.13 writeToFile(path as string) as boolean	589
– 6.4.1 class NSColorMBS	590
* 6.4.3 alternateSelectedControlColor as NSColorMBS	591
* 6.4.4 alternateSelectedControlTextColor as NSColorMBS	591
* 6.4.5 alternatingContentBackgroundColors as NSColorMBS()	592
* 6.4.6 blackColor as NSColorMBS	592
* 6.4.7 blendedColorWithFraction(alpha as Double, c as NSColorMBS) as NSColorMBS	592
* 6.4.8 blueColor as NSColorMBS	593
* 6.4.9 brownColor as NSColorMBS	593
* 6.4.10 CGColorHandle as Integer	593
* 6.4.11 clearColor as NSColorMBS	594
* 6.4.12 colorFromPasteboard as NSColorMBS	594
* 6.4.13 colorNamed(colorName as String) as NSColorMBS	594
* 6.4.14 colorSpace as NSColorSpaceMBS	594
* 6.4.15 colorUsingColorSpace(colorSpace as NSColorSpaceMBS) as NSColorMBS	595
* 6.4.16 colorUsingColorSpaceName(colorSpace as string) as NSColorMBS	595
* 6.4.17 colorUsingType(type as Integer) as NSColorMBS	596
* 6.4.18 colorWithAlphaComponent(alpha as Double) as NSColorMBS	596
* 6.4.19 colorWithCalibratedHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0) as NSColorMBS	596
* 6.4.20 colorWithCalibratedRGB(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS	597
* 6.4.21 colorWithCalibratedWhite(white as Double, alpha as Double=1.0) as NSColorMBS	597
* 6.4.22 colorWithCatalogName(listName as String, colorName as String) as NSColorMBS	598
* 6.4.23 colorWithCGColor(CGColorHandle as Integer) as NSColorMBS	598
* 6.4.24 colorWithColorSpace(ColorSpace as NSColorSpaceMBS, components() as Double) as NSColorMBS	598
* 6.4.25 colorWithColorSpace(ColorSpace as NSColorSpaceMBS, paramarray components as Double) as NSColorMBS	599

* 6.4.26	colorWithColorSpaceHSV(ColorSpace as NSColorSpaceMBS, hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0) as NSColorMBS	599
* 6.4.27	colorWithDeviceCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double=1.0) as NSColorMBS	600
* 6.4.28	colorWithDeviceHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0) as NSColorMBS	600
* 6.4.29	colorWithDeviceRGB(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS	601
* 6.4.30	colorWithDeviceWhite(white as Double, alpha as Double=1.0) as NSColorMBS	601
* 6.4.31	colorWithDisplayP3(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS	601
* 6.4.32	colorWithGenericGamma22White(white as Double, alpha as Double=1.0) as NSColorMBS	602
* 6.4.33	colorWithHSV(hue as double, saturation as double, brightness as double, alpha as double=1.0) as NSColorMBS	602
* 6.4.34	colorWithPatternImage(image as Variant) as NSColorMBS	602
* 6.4.35	colorWithRGB(red as double, green as double, blue as double, alpha as double=1.0) as NSColorMBS	603
* 6.4.36	colorWithSRGB(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS	603
* 6.4.37	colorWithSystemEffect(systemEffect as Integer) as NSColorMBS	603
* 6.4.38	colorWithWhite(white as double, alpha as double=1.0) as NSColorMBS	604
* 6.4.39	Components as Double()	604
* 6.4.40	Constructor(c as color)	604
* 6.4.41	Constructor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)	605
* 6.4.42	controlAccentColor as NSColorMBS	605
* 6.4.43	controlAlternatingRowBackgroundColors as NSColorMBS()	605
* 6.4.44	controlBackgroundColor as NSColorMBS	605
* 6.4.45	controlColor as NSColorMBS	606
* 6.4.46	controlDarkShadowColor as NSColorMBS	606
* 6.4.47	controlHighlightColor as NSColorMBS	607
* 6.4.48	controlLightHighlightColor as NSColorMBS	607
* 6.4.49	controlShadowColor as NSColorMBS	608
* 6.4.50	controlTextColor as NSColorMBS	608
* 6.4.51	currentControlTint as Integer	608
* 6.4.52	cyanColor as NSColorMBS	608
* 6.4.53	darkGrayColor as NSColorMBS	609
* 6.4.54	disabledControlTextColor as NSColorMBS	609
* 6.4.55	findHighlightColor as NSColorMBS	609
* 6.4.56	getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double)	610
* 6.4.57	getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double, byref alpha as Double)	610

* 6.4.58 getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double)	611
* 6.4.59 getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double, byref alpha as Double)	611
* 6.4.60 getRGB(byref red as Double, byref green as Double, byref blue as Double)	612
* 6.4.61 getRGB(byref red as Double, byref green as Double, byref blue as Double, byref alpha as Double)	613
* 6.4.62 getWhite(byref white as Double)	613
* 6.4.63 getWhite(byref white as Double, byref alpha as Double)	614
* 6.4.64 grayColor as NSColorMBS	614
* 6.4.65 greenColor as NSColorMBS	615
* 6.4.66 gridColor as NSColorMBS	615
* 6.4.67 headerColor as NSColorMBS	615
* 6.4.68 headerTextColor as NSColorMBS	616
* 6.4.69 highlightColor as NSColorMBS	616
* 6.4.70 highlightWithLevel(level as Double) as NSColorMBS	616
* 6.4.71 keyboardFocusIndicatorColor as NSColorMBS	617
* 6.4.72 knobColor as NSColorMBS	617
* 6.4.73 labelColor as NSColorMBS	618
* 6.4.74 lightGrayColor as NSColorMBS	618
* 6.4.75 linkColor as NSColorMBS	618
* 6.4.76 magentaColor as NSColorMBS	618
* 6.4.77 orangeColor as NSColorMBS	619
* 6.4.78 patternImage as Variant	619
* 6.4.79 placeholderTextColor as NSColorMBS	619
* 6.4.80 purpleColor as NSColorMBS	620
* 6.4.81 quaternaryLabelColor as NSColorMBS	620
* 6.4.82 redColor as NSColorMBS	620
* 6.4.83 scrollBarColor as NSColorMBS	620
* 6.4.84 scrubberTexturedBackgroundColor as NSColorMBS	621
* 6.4.85 secondaryLabelColor as NSColorMBS	621
* 6.4.86 secondarySelectedControlColor as NSColorMBS	621
* 6.4.87 selectedContentBackgroundColor as NSColorMBS	621
* 6.4.88 selectedControlColor as NSColorMBS	622
* 6.4.89 selectedControlTextColor as NSColorMBS	622
* 6.4.90 selectedKnobColor as NSColorMBS	622
* 6.4.91 selectedMenuItemColor as NSColorMBS	623
* 6.4.92 selectedMenuItemTextColor as NSColorMBS	623
* 6.4.93 selectedTextBackgroundColor as NSColorMBS	624
* 6.4.94 selectedTextColor as NSColorMBS	624
* 6.4.95 separatorColor as NSColorMBS	624
* 6.4.96 shadowColor as NSColorMBS	624

* 6.4.97 shadowWithLevel(level as Double) as NSColorMBS	625
* 6.4.98 systemBlueColor as NSColorMBS	625
* 6.4.99 systemBrownColor as NSColorMBS	625
* 6.4.100 systemGrayColor as NSColorMBS	626
* 6.4.101 systemGreenColor as NSColorMBS	626
* 6.4.102 systemIndigoColor as NSColorMBS	626
* 6.4.103 systemOrangeColor as NSColorMBS	627
* 6.4.104 systemPinkColor as NSColorMBS	627
* 6.4.105 systemPurpleColor as NSColorMBS	627
* 6.4.106 systemRedColor as NSColorMBS	627
* 6.4.107 systemTealColor as NSColorMBS	627
* 6.4.108 systemYellowColor as NSColorMBS	627
* 6.4.109 tertiaryLabelColor as NSColorMBS	628
* 6.4.110 textBackgroundColor as NSColorMBS	628
* 6.4.111 textColor as NSColorMBS	628
* 6.4.112 underPageBackgroundColor as NSColorMBS	629
* 6.4.113 unemphasizedSelectedContentBackgroundColor as NSColorMBS	629
* 6.4.114 unemphasizedSelectedTextBackgroundColor as NSColorMBS	629
* 6.4.115 unemphasizedSelectedTextColor as NSColorMBS	629
* 6.4.116 whiteColor as NSColorMBS	629
* 6.4.117 windowBackgroundColor as NSColorMBS	630
* 6.4.118 windowFrameColor as NSColorMBS	630
* 6.4.119 windowFrameTextColor as NSColorMBS	630
* 6.4.120 writeToPasteboard	631
* 6.4.121 yellowColor as NSColorMBS	631
* 6.4.123 alphaComponent as Double	631
* 6.4.124 blackComponent as Double	632
* 6.4.125 blueComponent as Double	632
* 6.4.126 brightnessComponent as Double	632
* 6.4.127 catalogNameComponent as string	633
* 6.4.128 colorNameComponent as string	633
* 6.4.129 colorSpaceName as string	633
* 6.4.130 colorValue as color	633
* 6.4.131 cyanComponent as Double	634
* 6.4.132 description as string	634
* 6.4.133 greenComponent as Double	634
* 6.4.134 Handle as Integer	635
* 6.4.135 hueComponent as Double	635
* 6.4.136 localizedCatalogNameComponent as string	635
* 6.4.137 localizedColorNameComponent as string	635
* 6.4.138 magentaComponent as Double	636
* 6.4.139 numberOfComponents as Integer	636

* 6.4.140 redComponent as Double	636
* 6.4.141 saturationComponent as Double	637
* 6.4.142 Type as Integer	637
* 6.4.143 whiteComponent as Double	638
* 6.4.144 yellowComponent as Double	638
– 6.5.1 class NSColorSamplerMBS	640
* 6.5.3 Available as Boolean	640
* 6.5.4 Constructor	640
* 6.5.5 Show	640
* 6.5.7 Handle as Integer	641
* 6.5.9 Completed(selectedColor as NSColorMBS)	641
– 6.6.1 class NSColorSpaceMBS	642
* 6.6.3 adobeRGB1998ColorSpace as NSColorSpaceMBS	642
* 6.6.4 availableColorSpacesWithModel(Model as Integer) as NSColorSpaceMBS()	643
* 6.6.5 CGColorSpaceHandle as Integer	643
* 6.6.6 colorSpaceForColorSpaceName(name as string) as NSColorSpaceMBS	644
* 6.6.7 ColorSpaceWithCGColorSpace(CGColorSpaceHandle as Integer) as NSColorSpaceMBS	644
* 6.6.8 ColorSpaceWithColorSyncProfile(ColorSyncProfileHandle as Integer) as NSColorSpaceMBS	644
* 6.6.9 ColorSpaceWithICCProfileData(File as FolderItem) as NSColorSpaceMBS	645
* 6.6.10 ColorSpaceWithICCProfileData(ICCProfileData as Memoryblock) as NSColorSpaceMBS	645
* 6.6.11 colorSyncProfileHandle as Integer	645
* 6.6.12 Constructor(ICCProfileData as Memoryblock)	646
* 6.6.13 deviceCMYKColorSpace as NSColorSpaceMBS	646
* 6.6.14 deviceGrayColorSpace as NSColorSpaceMBS	647
* 6.6.15 deviceRGBColorSpace as NSColorSpaceMBS	647
* 6.6.16 genericCMYKColorSpace as NSColorSpaceMBS	647
* 6.6.17 genericGamma22GrayColorSpace as NSColorSpaceMBS	648
* 6.6.18 genericGrayColorSpace as NSColorSpaceMBS	648
* 6.6.19 genericRGBColorSpace as NSColorSpaceMBS	648
* 6.6.20 ICCProfileData as Memoryblock	649
* 6.6.21 initWithCGColorSpace(CGColorSpaceHandle as Integer)	649
* 6.6.22 initWithColorSyncProfile(ColorSyncProfileHandle as Integer)	649
* 6.6.23 sRGBColorSpace as NSColorSpaceMBS	650
* 6.6.25 colorSpaceModel as Integer	650
* 6.6.26 colorSpaceName as string	651
* 6.6.27 description as string	651
* 6.6.28 File as FolderItem	651
* 6.6.29 Handle as Integer	651
* 6.6.30 localizedName as string	652
* 6.6.31 numberOfColorComponents as Integer	652

• 4 Cocoa	147
– 4.5.1 class NSCursorMBS	194
* 4.5.3 arrowCursor as NSCursorMBS	195
* 4.5.4 closedHandCursor as NSCursorMBS	195
* 4.5.5 Constructor(image as NSImageMBS, foregroundColorHint as NSColorMBS, background-ColorHint as NSColorMBS, HotSpotX as Double, HotSpotY as Double)	196
* 4.5.6 Constructor(image as NSImageMBS, HotSpotX as Double, HotSpotY as Double)	196
* 4.5.7 contextualMenuCursor as NSCursorMBS	196
* 4.5.8 crosshairCursor as NSCursorMBS	196
* 4.5.9 currentCursor as NSCursorMBS	197
* 4.5.10 currentSystemCursor as NSCursorMBS	197
* 4.5.11 disappearingItemCursor as NSCursorMBS	198
* 4.5.12 dragCopyCursor as NSCursorMBS	198
* 4.5.13 dragLinkCursor as NSCursorMBS	198
* 4.5.14 hide	199
* 4.5.15 hotSpotX as Double	199
* 4.5.16 hotSpotY as Double	200
* 4.5.17 IBeamCursor as NSCursorMBS	200
* 4.5.18 IBeamCursorForVerticalLayout as NSCursorMBS	200
* 4.5.19 image as NSImageMBS	201
* 4.5.20 isSetOnMouseEntered as boolean	201
* 4.5.21 isSetOnMouseExited as boolean	201
* 4.5.22 mouseEntered(e as NSEventMBS)	202
* 4.5.23 mouseExited(e as NSEventMBS)	202
* 4.5.24 openHandCursor as NSCursorMBS	202
* 4.5.25 operationNotAllowedCursor as NSCursorMBS	203
* 4.5.26 pointingHandCursor as NSCursorMBS	203
* 4.5.27 pop	203
* 4.5.28 pop	204
* 4.5.29 push	204
* 4.5.30 resizeDownCursor as NSCursorMBS	204
* 4.5.31 resizeLeftCursor as NSCursorMBS	204
* 4.5.32 resizeLeftRightCursor as NSCursorMBS	205
* 4.5.33 resizeRightCursor as NSCursorMBS	205
* 4.5.34 resizeUpCursor as NSCursorMBS	206
* 4.5.35 resizeUpDownCursor as NSCursorMBS	206
* 4.5.36 ringCursorWithDiameter(diameter as Double) as NSCursorMBS	207
* 4.5.37 set	207
* 4.5.38 setHiddenUntilMouseMoves(value as boolean)	207
* 4.5.39 setOnMouseEntered(flag as boolean)	207
* 4.5.40 setOnMouseExited(flag as boolean)	208

	29
* 4.5.41 unhide	208
* 4.5.43 Handle as Integer	208
– 4.6.1 class NSDateComponentsMBS	209
* 4.6.3 Constructor	209
* 4.6.4 copy as NSDateComponentsMBS	209
* 4.6.5 isValidDateInCalendar(calendar as NSCalendarMBS) as Boolean	209
* 4.6.6 NSUndefinedDateComponent as Integer	210
* 4.6.7 Print	210
* 4.6.9 calendar as NSCalendarMBS	210
* 4.6.10 date as date	210
* 4.6.11 dateTime as DateTime	211
* 4.6.12 day as Integer	211
* 4.6.13 description as string	211
* 4.6.14 era as Integer	211
* 4.6.15 Handle as Integer	211
* 4.6.16 hour as Integer	211
* 4.6.17 isLeapMonth as Boolean	212
* 4.6.18 isValidDate as Boolean	212
* 4.6.19 minute as Integer	212
* 4.6.20 month as Integer	212
* 4.6.21 quarter as Integer	212
* 4.6.22 second as Integer	213
* 4.6.23 timeZone as NSTimeZoneMBS	213
* 4.6.24 week as Integer	213
* 4.6.25 weekday as Integer	213
* 4.6.26 weekdayOrdinal as Integer	213
* 4.6.27 weekOfMonth as Integer	214
* 4.6.28 weekOfYear as Integer	214
* 4.6.29 year as Integer	214
* 4.6.30 yearForWeekOfYear as Integer	214
– 4.7.1 class NSDirectoryEnumeratorMBS	215
* 4.7.3 Constructor(folder as folderitem)	215
* 4.7.4 Constructor(path as string)	216
* 4.7.5 Destructor	216
* 4.7.6 directoryAttributes as dictionary	217
* 4.7.7 fileAttributes as dictionary	217
* 4.7.8 level as Integer	217
* 4.7.9 nextFile as folderitem	218
* 4.7.10 NSFileAppendOnly as string	218
* 4.7.11 NSFileBusy as string	218
* 4.7.12 NSFileCreationDate as string	219

* 4.7.13 NSFileDeviceIdentifier as string	219
* 4.7.14 NSFileExtensionHidden as string	219
* 4.7.15 NSFileGroupOwnerAccountID as string	219
* 4.7.16 NSFileGroupOwnerAccountName as string	219
* 4.7.17 NSFileHFSCreatorCode as string	220
* 4.7.18 NSFileHFSTypeCode as string	220
* 4.7.19 NSFileImmutable as string	220
* 4.7.20 NSFileModificationDate as string	220
* 4.7.21 NSFileOwnerAccountID as string	220
* 4.7.22 NSFileOwnerAccountName as string	220
* 4.7.23 NSFilePosixPermissions as string	221
* 4.7.24 NSFileReferenceCount as string	221
* 4.7.25 NSFileSize as string	221
* 4.7.26 NSFileSystemFileNumber as string	222
* 4.7.27 NSFileSystemFreeNodes as string	222
* 4.7.28 NSFileSystemFreeSize as string	222
* 4.7.29 NSFileSystemNodes as string	222
* 4.7.30 NSFileSystemNumber as string	222
* 4.7.31 NSFileSystemSize as string	223
* 4.7.32 NSFileType as string	223
* 4.7.33 NSFileTypeBlockSpecial as string	223
* 4.7.34 NSFileTypeCharacterSpecial as string	223
* 4.7.35 NSFileTypeDirectory as string	223
* 4.7.36 NSFileTypeRegular as string	223
* 4.7.37 NSFileTypeSocket as string	224
* 4.7.38 NSFileTypeSymbolicLink as string	224
* 4.7.39 NSFileTypeUnknown as string	224
* 4.7.40 Path as string	224
* 4.7.41 skipDescendents	224

	31
• 14 Notifications	1097
– 14.1.1 class NSDistributedNotificationCenterMBS	1097
* 14.1.3 addObserver(observer as NSNotificationObserverMBS, name as string, theObject as Variant, suspensionBehavior as Integer)	1098
* 14.1.4 Constructor	1098
* 14.1.5 defaultCenter as NSDistributedNotificationCenterMBS	1098
* 14.1.6 notificationCenterForType(name as string) as NSDistributedNotificationCenterMBS	1099
* 14.1.7 NSLocalNotificationCenterType as string	1099
* 14.1.8 postNotificationName(name as string, theObject as string, userInfo as dictionary, deliverImmediately as boolean)	1099
* 14.1.9 postNotificationName(name as string, theObject as string, userInfo as dictionary, options as UInt32)	1100
* 14.1.11 suspended as boolean	1100

• 4 Cocoa	147
– 4.8.1 class NSEnumeratorMBS	225
* 4.8.3 allObjects as Variant()	225
* 4.8.4 Constructor	226
* 4.8.5 nextObject as Variant	226
* 4.8.7 Handle as Integer	226

	33
• 6 Cocoa Drawing	551
– 6.7.1 class NSEPSImageRepMBS	654
* 6.7.3 boundingBox as NSRectMBS	654
* 6.7.4 Constructor(data as Memoryblock)	654
* 6.7.5 EPSRepresentation as Memoryblock	654
* 6.7.6 imageRepWithData(data as Memoryblock) as NSEPSImageRepMBS	655
* 6.7.7 prepareGState	655
* 6.7.9 pdfImage as NSPDFImageRepMBS	655

• 4 Cocoa	147
– 4.9.1 class NSEventMBS	227
* 4.9.3 allTouches as NSTouchMBS()	227
* 4.9.4 Constructor	228
* 4.9.5 doubleClickInterval as Double	228
* 4.9.6 eventWithCGEvent(CGEventRef as Integer) as NSEventMBS	228
* 4.9.7 isMouseCoalescingEnabled as boolean	228
* 4.9.8 keyEvent(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, characters as String, charactersIgnoringModifiers as String, isARepet as boolean, keyCode as Integer) as NSEventMBS	228
* 4.9.9 keyRepeatDelay as Double	229
* 4.9.10 keyRepeatInterval as Double	229
* 4.9.11 modifierFlagsGlobal as UInt32	229
* 4.9.12 mouseEvent(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, eventNumber as Integer, clickCount as Integer, pressure as Single) as NSEventMBS	230
* 4.9.13 mouseLocation as NSPointMBS	230
* 4.9.14 otherEvent(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, SubType as Integer, Data1 as Integer, Data2 as Integer) as NSEventMBS	231
* 4.9.15 pressedMouseButtons as UInt32	231
* 4.9.16 setMouseCoalescingEnabled(Value as boolean)	231
* 4.9.17 touchesForView(view as NSViewMBS) as NSTouchMBS()	232
* 4.9.18 touchesForWindow(win as NSWindowMBS) as NSTouchMBS()	232
* 4.9.20 absoluteX as Integer	232
* 4.9.21 absoluteY as Integer	232
* 4.9.22 absoluteZ as Integer	233
* 4.9.23 associatedEventsMask as Integer	233
* 4.9.24 buttonMask as Integer	233
* 4.9.25 buttonNumber as Integer	234
* 4.9.26 capabilityMask as Integer	234
* 4.9.27 CGEventRef as Integer	234
* 4.9.28 characters as string	235
* 4.9.29 charactersIgnoringModifiers as string	235
* 4.9.30 clickCount as Integer	235
* 4.9.31 data1 as Integer	236
* 4.9.32 data2 as Integer	236
* 4.9.33 deltaX as Double	236
* 4.9.34 deltaY as Double	236
* 4.9.35 deltaZ as Double	237
* 4.9.36 description as string	237

* 4.9.37 deviceID as Integer	237
* 4.9.38 eventNumber as Integer	237
* 4.9.39 Handle as Integer	238
* 4.9.40 hasPreciseScrollingDeltas as boolean	238
* 4.9.41 isARepet as boolean	238
* 4.9.42 isDirectionInvertedFromDevice as boolean	238
* 4.9.43 isEnteringProximity as boolean	239
* 4.9.44 keyCode as Integer	239
* 4.9.45 locationInWindow as NSPointMBS	239
* 4.9.46 magnification as Double	240
* 4.9.47 modifierFlags as Integer	240
* 4.9.48 pointingDeviceID as Integer	240
* 4.9.49 pointingDeviceSerialNumber as Integer	240
* 4.9.50 pointingDeviceType as Integer	241
* 4.9.51 pressure as Double	241
* 4.9.52 rotation as Double	241
* 4.9.53 scrollingDeltaX as Double	242
* 4.9.54 scrollingDeltaY as Double	242
* 4.9.55 stage as Integer	242
* 4.9.56 stageTransition as Double	243
* 4.9.57 subtype as Integer	243
* 4.9.58 systemTabletID as Integer	243
* 4.9.59 tabletID as Integer	243
* 4.9.60 tangentialPressure as Double	244
* 4.9.61 tilt as NSPointMBS	244
* 4.9.62 timestamp as Double	244
* 4.9.63 trackingNumber as Integer	245
* 4.9.64 type as Integer	245
* 4.9.65 uniqueID as UInt64	246
* 4.9.66 vendorID as Integer	246
* 4.9.67 vendorPointingDeviceType as Integer	246
* 4.9.68 window as NSWindowMBS	246
* 4.9.69 windowNumber as Integer	247

• 9 Cocoa Text	925
– 9.2.1 class NSFileWrapperMBS	984
* 9.2.3 addFileWrapper(child as NSFileWrapperMBS) as String	985
* 9.2.4 addRegularFileWithContents(Data as MemoryBlock, preferredFilename as string) as String	986
* 9.2.5 Constructor	986
* 9.2.6 initWithDirectoryWithFileWrappers(childrenByPreferredName as Dictionary) as NSFileWrapperMBS	986
* 9.2.7 initWithRegularFileWithContents(data as MemoryBlock) as NSFileWrapperMBS	987
* 9.2.8 initWithFile(File as FolderItem, Options as Integer, byref error as NSErrorMBS) as NSFileWrapperMBS	987
* 9.2.9 initWithSerializedRepresentation(data as MemoryBlock) as NSFileWrapperMBS	988
* 9.2.10 initWithURL(URL as string, Options as Integer, byref error as NSErrorMBS) as NSFileWrapperMBS	989
* 9.2.11 keyForFileWrapper(child as NSFileWrapperMBS) as String	989
* 9.2.12 matchesContentsOfFile(File as FolderItem) as Boolean	989
* 9.2.13 matchesContentsOfURL(URL as String) as Boolean	990
* 9.2.14 readFromFile(File as FolderItem, Options as Integer = 0, byref Error as NSErrorMBS) as Boolean	991
* 9.2.15 readFromURL(URL as String, Options as Integer = 0, byref Error as NSErrorMBS) as Boolean	991
* 9.2.16 removeFileWrapper(child as NSFileWrapperMBS)	992
* 9.2.17 writeToFile(File as FolderItem, Options as Integer = 0, originalContentsURL as FolderItem = nil, byref Error as NSErrorMBS) as Boolean	992
* 9.2.18 writeToURL(URL as String, Options as Integer = 0, originalContentsURL as String = "", byref Error as NSErrorMBS) as Boolean	993
* 9.2.20 Directory as Boolean	994
* 9.2.21 fileAttributes as Dictionary	994
* 9.2.22 filename as String	994
* 9.2.23 fileWrappers as Dictionary	995
* 9.2.24 Handle as Integer	995
* 9.2.25 icon as NSImageMBS	996
* 9.2.26 preferredFilename as String	996
* 9.2.27 RegularFile as Boolean	997
* 9.2.28 regularFileContents as MemoryBlock	997
* 9.2.29 serializedRepresentation as MemoryBlock	997
* 9.2.30 SymbolicLink as Boolean	998
* 9.2.31 symbolicLinkDestinationURL as String	998

	37
• 4 Cocoa	147
– 4.10.1 class NSFontDescriptorMBS	249
* 4.10.3 Constructor(AttributesDic as Dictionary)	249
* 4.10.4 copy as NSFontDescriptorMBS	249
* 4.10.5 fontAttributes as Dictionary	250
* 4.10.6 fontDescriptorByAddingAttributes(AttributesDic as Dictionary) as NSFontDescriptorMBS	250
* 4.10.7 fontDescriptorWithFace(newFace as string) as NSFontDescriptorMBS	250
* 4.10.8 fontDescriptorWithFamily(newFamily as string) as NSFontDescriptorMBS	250
* 4.10.9 fontDescriptorWithFontAttributes(AttributesDic as Dictionary) as NSFontDescriptorMBS	250
* 4.10.10 fontDescriptorWithMatrix(matrix as Variant) as NSFontDescriptorMBS	251
* 4.10.11 fontDescriptorWithName(fontName as string, matrix as Variant) as NSFontDescriptorMBS	251
* 4.10.12 fontDescriptorWithName(fontName as string, size as Double) as NSFontDescriptorMBS	252
* 4.10.13 fontDescriptorWithSize(newPointSize as Double) as NSFontDescriptorMBS	252
* 4.10.14 fontDescriptorWithSymbolicTraits(SymbolicTraits as Integer) as NSFontDescriptorMBS	253
* 4.10.15 matchingFontDescriptorsWithMandatoryKeys as NSFontDescriptorMBS()	253
* 4.10.16 matchingFontDescriptorsWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS()	254
* 4.10.17 matchingFontDescriptorWithMandatoryKeys as NSFontDescriptorMBS	254
* 4.10.18 matchingFontDescriptorWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS	255
* 4.10.19 matrix as Variant	255
* 4.10.20 NSFontCascadeListAttribute as string	255
* 4.10.21 NSFontCharacterSetAttribute as string	255
* 4.10.22 NSFontColorAttribute as string	256
* 4.10.23 NSFontFaceAttribute as string	256
* 4.10.24 NSFontFamilyAttribute as string	256
* 4.10.25 NSFontFeatureSelectorIdentifierKey as string	256
* 4.10.26 NSFontFeatureTypeIdentifierKey as string	256
* 4.10.27 NSFontFixedAdvanceAttribute as string	257
* 4.10.28 NSFontMatrixAttribute as string	257
* 4.10.29 NSFontNameAttribute as string	257
* 4.10.30 NSFontSizeAttribute as string	257
* 4.10.31 NSFontSlantTrait as string	257
* 4.10.32 NSFontSymbolicTrait as string	258
* 4.10.33 NSFontTraitsAttribute as string	258
* 4.10.34 NSFontVariationAttribute as string	258
* 4.10.35 NSFontVariationAxisDefaultValueKey as string	258

* 4.10.36	NSFontVariationAxisIdentifierKey as string	258
* 4.10.37	NSFontVariationAxisMaximumValueKey as string	259
* 4.10.38	NSFontVariationAxisMinimumValueKey as string	259
* 4.10.39	NSFontVariationAxisNameKey as string	259
* 4.10.40	NSFontVisibleNameAttribute as string	259
* 4.10.41	NSFontWeightTrait as string	259
* 4.10.42	NSFontWidthTrait as string	260
* 4.10.43	pointSize as Double	260
* 4.10.44	postscriptName as string	260
* 4.10.45	symbolicTraits as Integer	260
* 4.10.46	variantForKey(key as string) as Variant	261
* 4.10.48	Handle as Integer	261
– 4.11.1	class NSFontMBS	264
* 4.11.3	advancementForGlyph(aGlyph as Integer) as NSSizeMBS	264
* 4.11.4	boldSystemFontOfSize(size as Double) as NSFontMBS	265
* 4.11.5	boundingRectForGlyph(aGlyph as Integer) as NSRectMBS	265
* 4.11.6	Constructor	265
* 4.11.7	controlContentFontOfSize(size as Double) as NSFontMBS	265
* 4.11.8	file as folderitem	266
* 4.11.9	fontDescriptor as NSFontDescriptorMBS	266
* 4.11.10	fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, fontSize as Double) as NSFontMBS	266
* 4.11.11	fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, TextTransform as Variant) as NSFontMBS	267
* 4.11.12	fontWithName(fontName as string, fontSize as Double) as NSFontMBS	267
* 4.11.13	glyphWithName(name as string) as UInt32	267
* 4.11.14	labelFontOfSize(size as Double) as NSFontMBS	268
* 4.11.15	labelFontSize as Double	268
* 4.11.16	menuBarFontOfSize(size as Double) as NSFontMBS	268
* 4.11.17	menuFontOfSize(size as Double) as NSFontMBS	269
* 4.11.18	messageFontOfSize(size as Double) as NSFontMBS	269
* 4.11.19	monospacedDigitSystemFontOfSize(fontSize as double, weight as double) as NSFontMBS	269
* 4.11.20	monospacedSystemFontOfSize(fontSize as double, weight as double) as NSFontMBS	270
* 4.11.21	paletteFontOfSize(size as Double) as NSFontMBS	270
* 4.11.22	screenFontWithRenderingMode(renderingMode as Integer) as NSFontMBS	271
* 4.11.23	setUserFixedPitchFont(font as NSFontMBS)	271
* 4.11.24	setUserFont(font as NSFontMBS)	271
* 4.11.25	smallSystemFontSize as Double	271
* 4.11.26	systemFontOfSize(size as Double) as NSFontMBS	271
* 4.11.27	systemFontSize as Double	272

* 4.11.28	systemFontSizeForControlSize(controlSize as Integer) as Double	272
* 4.11.29	titleBarFontOfSize(size as Double) as NSFontMBS	272
* 4.11.30	toolTipsFontOfSize(size as Double) as NSFontMBS	273
* 4.11.31	userFixedPitchFontOfSize(size as Double) as NSFontMBS	273
* 4.11.32	userFontOfSize(size as Double) as NSFontMBS	274
* 4.11.34	ascender as Double	274
* 4.11.35	boundingRectForFont as NSRectMBS	274
* 4.11.36	capHeight as Double	274
* 4.11.37	coveredCharacterSet as Variant	275
* 4.11.38	descender as Double	275
* 4.11.39	description as string	275
* 4.11.40	displayName as string	276
* 4.11.41	familyName as string	276
* 4.11.42	fontName as string	276
* 4.11.43	Handle as Integer	277
* 4.11.44	isFixedPitch as boolean	277
* 4.11.45	italicAngle as Double	277
* 4.11.46	leading as Double	277
* 4.11.47	maximumAdvancement as NSSizeMBS	277
* 4.11.48	mostCompatibleStringEncoding as Integer	278
* 4.11.49	numberOfGlyphs as Integer	278
* 4.11.50	pointSize as Double	278
* 4.11.51	printerFont as NSFontMBS	279
* 4.11.52	renderingMode as Integer	279
* 4.11.53	screenFont as NSFontMBS	279
* 4.11.54	textTransform as Variant	279
* 4.11.55	underlinePosition as Double	280
* 4.11.56	underlineThickness as Double	280
* 4.11.57	xHeight as Double	280
– 4.12.1	class NSHelpManagerMBS	282
* 4.12.3	Constructor	282
* 4.12.4	findString(query as string, book as string)	282
* 4.12.5	isContextHelpModeActive as boolean	282
* 4.12.6	NSContextHelpModeDidActivateNotification as string	283
* 4.12.7	NSContextHelpModeDidDeactivateNotification as string	283
* 4.12.8	openHelpAnchor(anchor as string, book as string)	283
* 4.12.9	registerBooksInBundle(bundle as NSBundleMBS) as boolean	283
* 4.12.10	setContextHelpModeActive(active as boolean)	284
* 4.12.12	Handle as Integer	284

• 6 Cocoa Drawing	551
– 6.8.1 class NSImageMBS	657
* 6.8.3 addRepresentation(img as NSImageRepMBS)	658
* 6.8.4 BMPRepresentation as Memoryblock	658
* 6.8.5 BMPRepresentationMT as Memoryblock	659
* 6.8.6 cancelIncrementalLoad	659
* 6.8.7 canInitWithPasteboard as boolean	659
* 6.8.8 Constructor	660
* 6.8.9 Constructor(data as Memoryblock)	660
* 6.8.10 Constructor(file as folderitem)	660
* 6.8.11 Constructor(image as Picture, mask as picture = nil)	661
* 6.8.12 Constructor(width as Double, height as Double)	661
* 6.8.13 CopyMask as picture	662
* 6.8.14 CopyPicture(CGColorSpace as Variant = nil, BackgroundColor as NSColorMBS = nil) as picture	662
* 6.8.15 CopyPictureRect(x as Integer, y as Integer, w as Integer, h as Integer, CGColorSpace as Variant = nil, BackgroundColor as NSColorMBS = nil) as picture	663
* 6.8.16 CopyPictureWithAlpha as picture	663
* 6.8.17 CopyPictureWithAlphaRect(x as Integer, y as Integer, w as Integer, h as Integer) as picture	664
* 6.8.18 CopyPictureWithMask(CGColorSpace as Variant = nil) as picture	664
* 6.8.19 DrawIntoCGContextAtPoint(cgcontext as Integer, x as Double, y as Double, sx as Double, sy as Double, SourceW as Double, SourceH as Double, operation as Integer, fraction as Double) as boolean	665
* 6.8.20 DrawIntoCGContextAtRect(cgcontext as Integer, x as Double, y as Double, w as Double, h as Double, SourceX as Double, SourceY as Double, SourceW as Double, SourceH as Double, operation as Integer, fraction as Double) as boolean	665
* 6.8.21 GIFRepresentation as Memoryblock	667
* 6.8.22 GIFRepresentationMT as Memoryblock	667
* 6.8.23 imageByFadingToFraction(fraction as Double) as NSImageMBS	667
* 6.8.24 imageByScalingToSize(width as Double, height as Double) as NSImageMBS	668
* 6.8.25 imageByScalingToSize(width as Double, height as Double, fraction as Double) as NSImageMBS	669
* 6.8.26 imageByScalingToSize(width as Double, height as Double, fraction as Double, flip as boolean, proportionally as boolean) as NSImageMBS	670
* 6.8.27 imageFileTypes as string()	671
* 6.8.28 imageNamed(name as string) as NSImageMBS	671
* 6.8.29 imagePasteboardTypes as string()	672
* 6.8.30 imageTypes as string()	673
* 6.8.31 imageUnfilteredFileTypes as string()	673
* 6.8.32 imageUnfilteredPasteboardTypes as string()	674
* 6.8.33 imageUnfilteredTypes as string()	674

* 6.8.34	initWithCGImage(CGImage as Variant, width as Double = 0, height as Double = 0) as NSImageMBS	674
* 6.8.35	initWithContentsOfFile(file as folderitem) as NSImageMBS	675
* 6.8.36	initWithContentsOfFileMT(file as folderitem) as NSImageMBS	676
* 6.8.37	initWithContentsOfPath(path as string) as NSImageMBS	676
* 6.8.38	initWithContentsOfPathMT(path as string) as NSImageMBS	677
* 6.8.39	initWithContentsOfURL(URL as string) as NSImageMBS	677
* 6.8.40	initWithContentsOfURLMT(URL as string) as NSImageMBS	678
* 6.8.41	initWithData(data as memoryblock) as NSImageMBS	678
* 6.8.42	initWithData(data as string) as NSImageMBS	679
* 6.8.43	initWithDataMT(data as memoryblock) as NSImageMBS	679
* 6.8.44	initWithDataMT(data as string) as NSImageMBS	680
* 6.8.45	initWithHandle(Handle as Integer) as NSImageMBS	680
* 6.8.46	initWithSymbolConfiguration(configuration as NSImageSymbolConfigurationMBS) as NSImageMBS	680
* 6.8.47	initWithSystemSymbolName(name as string, accessibilityDescription as string = "") as NSImageMBS	681
* 6.8.48	initWithTintColor(tintColor as NSColorMBS) as NSImageMBS	681
* 6.8.49	initWithContentsOfURL(file as folderitem) as boolean	681
* 6.8.50	initWithContentsOfURL(url as string) as boolean	682
* 6.8.51	initWithData(data as Memoryblock) as boolean	682
* 6.8.52	initWithDataIgnoringOrientation(data as Memoryblock) as boolean	682
* 6.8.53	initWithIconRef(IconHandle as Integer) as boolean	682
* 6.8.54	initWithPasteboard as boolean	682
* 6.8.55	initWithPicture(img as picture, mask as picture = nil) as boolean	683
* 6.8.56	initWithSize(width as Double, height as Double) as boolean	683
* 6.8.57	JPEGRepresentation as Memoryblock	683
* 6.8.58	JPEGRepresentationMT as Memoryblock	684
* 6.8.59	JPEGRepresentationWithCompressionFactor(factor as Double) as Memoryblock	685
* 6.8.60	JPEGRepresentationWithCompressionFactorMT(factor as Double) as Memoryblock	685
* 6.8.61	NSImageHintUserInterfaceLayoutDirection as string	686
* 6.8.62	NSImageNameActionTemplate as string	686
* 6.8.63	NSImageNameAddTemplate as string	687
* 6.8.64	NSImageNameAdvanced as string	687
* 6.8.65	NSImageNameApplicationIcon as string	687
* 6.8.66	NSImageNameBluetoothTemplate as string	688
* 6.8.67	NSImageNameBonjour as string	688
* 6.8.68	NSImageNameBookmarksTemplate as string	688
* 6.8.69	NSImageNameCaution as string	688
* 6.8.70	NSImageNameColorPanel as string	688
* 6.8.71	NSImageNameColumnViewTemplate as string	689

* 6.8.72 NSImageNameComputer as string	689
* 6.8.73 NSImageNameDotMac as string	689
* 6.8.74 NSImageNameEnterFullScreenTemplate as string	690
* 6.8.75 NSImageNameEveryone as string	690
* 6.8.76 NSImageNameExitFullScreenTemplate as string	690
* 6.8.77 NSImageNameFlowLayoutTemplate as string	690
* 6.8.78 NSImageNameFolder as string	690
* 6.8.79 NSImageNameFolderBurnable as string	691
* 6.8.80 NSImageNameFolderSmart as string	691
* 6.8.81 NSImageNameFollowLinkFreestandingTemplate as string	691
* 6.8.82 NSImageNameFontPanel as string	691
* 6.8.83 NSImageNameGoBackTemplate as string	692
* 6.8.84 NSImageNameGoForwardTemplate as string	692
* 6.8.85 NSImageNameGoLeftTemplate as string	692
* 6.8.86 NSImageNameGoRightTemplate as string	692
* 6.8.87 NSImageNameHomeTemplate as string	692
* 6.8.88 NSImageNameIChatTheaterTemplate as string	693
* 6.8.89 NSImageNameIconViewTemplate as string	693
* 6.8.90 NSImageNameInfo as string	693
* 6.8.91 NSImageNameInvalidDataFreestandingTemplate as string	693
* 6.8.92 NSImageNameLeftFacingTriangleTemplate as string	694
* 6.8.93 NSImageNameListViewTemplate as string	694
* 6.8.94 NSImageNameLockLockedTemplate as string	694
* 6.8.95 NSImageNameLockUnlockedTemplate as string	694
* 6.8.96 NSImageNameMenuMixedStateTemplate as string	694
* 6.8.97 NSImageNameMenuOnStateTemplate as string	695
* 6.8.98 NSImageNameMobileMe as string	695
* 6.8.99 NSImageNameMultipleDocuments as string	695
* 6.8.100 NSImageNameNetwork as string	695
* 6.8.101 NSImageNamePathTemplate as string	696
* 6.8.102 NSImageNamePreferencesGeneral as string	696
* 6.8.103 NSImageNameQuickLookTemplate as string	697
* 6.8.104 NSImageNameRefreshFreestandingTemplate as string	697
* 6.8.105 NSImageNameRefreshTemplate as string	697
* 6.8.106 NSImageNameRemoveTemplate as string	697
* 6.8.107 NSImageNameRevealFreestandingTemplate as string	698
* 6.8.108 NSImageNameRightFacingTriangleTemplate as string	698
* 6.8.109 NSImageNameShareTemplate as string	698
* 6.8.110 NSImageNameSlideshowTemplate as string	698
* 6.8.111 NSImageNameSmartBadgeTemplate as string	698
* 6.8.112 NSImageNameStatusAvailable as string	699
* 6.8.113 NSImageNameStatusNone as string	699

* 6.8.114	NSImageNameStatusPartiallyAvailable as string	699
* 6.8.115	NSImageNameStatusUnavailable as string	699
* 6.8.116	NSImageNameStopProgressFreestandingTemplate as string	700
* 6.8.117	NSImageNameStopProgressTemplate as string	700
* 6.8.118	NSImageNameTouchBarAddDetailTemplate as string	700
* 6.8.119	NSImageNameTouchBarAddTemplate as string	700
* 6.8.120	NSImageNameTouchBarAlarmTemplate as string	701
* 6.8.121	NSImageNameTouchBarAudioInputMuteTemplate as string	701
* 6.8.122	NSImageNameTouchBarAudioInputTemplate as string	702
* 6.8.123	NSImageNameTouchBarAudioOutputMuteTemplate as string	702
* 6.8.124	NSImageNameTouchBarAudioOutputVolumeHighTemplate as string	702
* 6.8.125	NSImageNameTouchBarAudioOutputVolumeLowTemplate as string	703
* 6.8.126	NSImageNameTouchBarAudioOutputVolumeMediumTemplate as string	703
* 6.8.127	NSImageNameTouchBarAudioOutputVolumeOffTemplate as string	704
* 6.8.128	NSImageNameTouchBarBookmarksTemplate as string	704
* 6.8.129	NSImageNameTouchBarColorPickerFill as string	705
* 6.8.130	NSImageNameTouchBarColorPickerFont as string	705
* 6.8.131	NSImageNameTouchBarColorPickerStroke as string	705
* 6.8.132	NSImageNameTouchBarCommunicationAudioTemplate as string	706
* 6.8.133	NSImageNameTouchBarCommunicationVideoTemplate as string	706
* 6.8.134	NSImageNameTouchBarComposeTemplate as string	707
* 6.8.135	NSImageNameTouchBarDeleteTemplate as string	707
* 6.8.136	NSImageNameTouchBarDownloadTemplate as string	707
* 6.8.137	NSImageNameTouchBarEnterFullScreenTemplate as string	708
* 6.8.138	NSImageNameTouchBarExitFullScreenTemplate as string	708
* 6.8.139	NSImageNameTouchBarFastForwardTemplate as string	709
* 6.8.140	NSImageNameTouchBarFolderCopyToTemplate as string	709
* 6.8.141	NSImageNameTouchBarFolderMoveToTemplate as string	709
* 6.8.142	NSImageNameTouchBarFolderTemplate as string	710
* 6.8.143	NSImageNameTouchBarGetInfoTemplate as string	710
* 6.8.144	NSImageNameTouchBarGoBackTemplate as string	711
* 6.8.145	NSImageNameTouchBarGoDownTemplate as string	711
* 6.8.146	NSImageNameTouchBarGoForwardTemplate as string	712
* 6.8.147	NSImageNameTouchBarGoUpTemplate as string	712
* 6.8.148	NSImageNameTouchBarHistoryTemplate as string	712
* 6.8.149	NSImageNameTouchBarIconViewTemplate as string	713
* 6.8.150	NSImageNameTouchBarListViewTemplate as string	713
* 6.8.151	NSImageNameTouchBarMailTemplate as string	714
* 6.8.152	NSImageNameTouchBarNewFolderTemplate as string	714
* 6.8.153	NSImageNameTouchBarNewMessageTemplate as string	714
* 6.8.154	NSImageNameTouchBarOpenInBrowserTemplate as string	715
* 6.8.155	NSImageNameTouchBarPauseTemplate as string	715

* 6.8.156	NSImageNameTouchBarPlayheadTemplate as string	716
* 6.8.157	NSImageNameTouchBarPlayPauseTemplate as string	716
* 6.8.158	NSImageNameTouchBarPlayTemplate as string	716
* 6.8.159	NSImageNameTouchBarQuickLookTemplate as string	717
* 6.8.160	NSImageNameTouchBarRecordStartTemplate as string	717
* 6.8.161	NSImageNameTouchBarRecordStopTemplate as string	718
* 6.8.162	NSImageNameTouchBarRefreshTemplate as string	718
* 6.8.163	NSImageNameTouchBarRewindTemplate as string	719
* 6.8.164	NSImageNameTouchBarRotateLeftTemplate as string	719
* 6.8.165	NSImageNameTouchBarRotateRightTemplate as string	719
* 6.8.166	NSImageNameTouchBarSearchTemplate as string	720
* 6.8.167	NSImageNameTouchBarShareTemplate as string	720
* 6.8.168	NSImageNameTouchBarSidebarTemplate as string	721
* 6.8.169	NSImageNameTouchBarSkipAhead15SecondsTemplate as string	721
* 6.8.170	NSImageNameTouchBarSkipAhead30SecondsTemplate as string	721
* 6.8.171	NSImageNameTouchBarSkipAheadTemplate as string	722
* 6.8.172	NSImageNameTouchBarSkipBack15SecondsTemplate as string	722
* 6.8.173	NSImageNameTouchBarSkipBack30SecondsTemplate as string	723
* 6.8.174	NSImageNameTouchBarSkipBackTemplate as string	723
* 6.8.175	NSImageNameTouchBarSkipToEndTemplate as string	723
* 6.8.176	NSImageNameTouchBarSkipToStartTemplate as string	724
* 6.8.177	NSImageNameTouchBarSlideshowTemplate as string	724
* 6.8.178	NSImageNameTouchBarTagIconTemplate as string	725
* 6.8.179	NSImageNameTouchBarTextBoldTemplate as string	725
* 6.8.180	NSImageNameTouchBarTextBoxTemplate as string	726
* 6.8.181	NSImageNameTouchBarTextCenterAlignTemplate as string	726
* 6.8.182	NSImageNameTouchBarTextItalicTemplate as string	726
* 6.8.183	NSImageNameTouchBarTextJustifiedAlignTemplate as string	727
* 6.8.184	NSImageNameTouchBarTextLeftAlignTemplate as string	727
* 6.8.185	NSImageNameTouchBarTextListTemplate as string	728
* 6.8.186	NSImageNameTouchBarTextRightAlignTemplate as string	728
* 6.8.187	NSImageNameTouchBarTextStrikethroughTemplate as string	728
* 6.8.188	NSImageNameTouchBarTextUnderlineTemplate as string	729
* 6.8.189	NSImageNameTouchBarUserAddTemplate as string	729
* 6.8.190	NSImageNameTouchBarUserGroupTemplate as string	730
* 6.8.191	NSImageNameTouchBarUserTemplate as string	730
* 6.8.192	NSImageNameTouchBarVolumeDownTemplate as string	730
* 6.8.193	NSImageNameTouchBarVolumeUpTemplate as string	731
* 6.8.194	NSImageNameTrashEmpty as string	731
* 6.8.195	NSImageNameTrashFull as string	732
* 6.8.196	NSImageNameUser as string	732
* 6.8.197	NSImageNameUserAccounts as string	732

* 6.8.198	NSImageNameUserGroup as string	732
* 6.8.199	NSImageNameUserGuest as string	732
* 6.8.200	PNGRepresentation as Memoryblock	733
* 6.8.201	PNGRepresentationMT as Memoryblock	733
* 6.8.202	recache	734
* 6.8.203	removeRepresentation(img as NSImageRepMBS)	735
* 6.8.204	RepresentationHeight(index as Integer) as Integer	735
* 6.8.205	representations as NSImageRepMBS()	735
* 6.8.206	RepresentationWidth(index as Integer) as Integer	735
* 6.8.207	setName(value as String) as Boolean	735
* 6.8.208	setSize(width as Double, height as Double)	736
* 6.8.209	TIFFRepresentation as Memoryblock	736
* 6.8.210	TIFFRepresentationMT as Memoryblock	737
* 6.8.211	TIFFRepresentationUsingCompression(comp as Integer, factor as Double) as Memoryblock	738
* 6.8.212	TIFFRepresentationUsingCompressionMT(comp as Integer, factor as Double) as Memoryblock	738
* 6.8.214	accessibilityDescription as string	739
* 6.8.215	backgroundColor as NSColorMBS	739
* 6.8.216	cacheDepthMatchesImageDepth as Boolean	740
* 6.8.217	cacheMode as Integer	740
* 6.8.218	EXIFData as Dictionary	740
* 6.8.219	Handle as Integer	740
* 6.8.220	height as Double	740
* 6.8.221	isCachedSeparately as Boolean	741
* 6.8.222	isDataRetained as Boolean	741
* 6.8.223	isFlipped as Boolean	741
* 6.8.224	isTemplate as Boolean	741
* 6.8.225	isValid as Boolean	742
* 6.8.226	matchesOnMultipleResolution as Boolean	742
* 6.8.227	MaximumPixelHeight as Integer	743
* 6.8.228	MaximumPixelWidth as Integer	743
* 6.8.229	MinimumPixelHeight as Integer	743
* 6.8.230	MinimumPixelWidth as Integer	743
* 6.8.231	name as String	743
* 6.8.232	prefersColorMatch as Boolean	744
* 6.8.233	RepresentationsCount as Integer	744
* 6.8.234	scalesWhenResized as Boolean	744
* 6.8.235	size as NSSizeMBS	744
* 6.8.236	symbolConfiguration as NSImageSymbolConfigurationMBS	745
* 6.8.237	usesEPSONResolutionMismatch as Boolean	745
* 6.8.238	width as Double	745

– 6.9.1 class <code>NSImageRepMBS</code>	747
* 6.9.3 <code>canInitWithData(data as memoryblock)</code> as Boolean	747
* 6.9.4 Constructor	747
* 6.9.5 <code>setSize(width as Double, height as Double)</code>	748
* 6.9.7 <code>bitsPerSample</code> as Integer	748
* 6.9.8 <code>colorSpaceName</code> as String	748
* 6.9.9 <code>Handle</code> as Integer	749
* 6.9.10 <code>hasAlpha</code> as Boolean	749
* 6.9.11 <code>height</code> as Double	749
* 6.9.12 <code>isOpaque</code> as Boolean	749
* 6.9.13 <code>pixelsHigh</code> as Integer	750
* 6.9.14 <code>pixelsWide</code> as Integer	750
* 6.9.15 <code>size</code> as <code>NSSizeMBS</code>	750
* 6.9.16 <code>width</code> as Double	750
– 6.10.1 class <code>NSImageSymbolConfigurationMBS</code>	752
* 6.10.3 <code>available</code> as Boolean	753
* 6.10.4 <code>configurationByApplyingConfiguration(configuration as NSImageSymbolConfigurationMBS)</code> as <code>NSImageSymbolConfigurationMBS</code>	753
* 6.10.5 <code>configurationPreferringMulticolor</code> as <code>NSImageSymbolConfigurationMBS</code>	753
* 6.10.6 <code>configurationWithHierarchicalColor(hierarchicalColor as NSColorMBS)</code> as <code>NSImageSymbolConfigurationMBS</code>	753
* 6.10.7 <code>configurationWithPaletteColors(paletteColors() as NSColorMBS)</code> as <code>NSImageSymbolConfigurationMBS</code>	754
* 6.10.8 <code>configurationWithPointSize(pointSize as Double, weight as Double)</code> as <code>NSImageSymbolConfigurationMBS</code>	754
* 6.10.9 <code>configurationWithPointSize(pointSize as Double, weight as Double, SymbolScale as Integer)</code> as <code>NSImageSymbolConfigurationMBS</code>	754
* 6.10.10 <code>configurationWithScale(SymbolScale as Integer)</code> as <code>NSImageSymbolConfigurationMBS</code>	755
* 6.10.11 <code>configurationWithTextStyle(TextStyle as Integer)</code> as <code>NSImageSymbolConfigurationMBS</code>	755
* 6.10.12 <code>configurationWithTextStyle(TextStyle as Integer, SymbolScale as Integer)</code> as <code>NSImageSymbolConfigurationMBS</code>	755
* 6.10.13 Constructor	755
* 6.10.14 <code>NSFontTextStyleBody</code> as String	756
* 6.10.15 <code>NSFontTextStyleCallout</code> as String	756
* 6.10.16 <code>NSFontTextStyleCaption1</code> as String	756
* 6.10.17 <code>NSFontTextStyleCaption2</code> as String	756
* 6.10.18 <code>NSFontTextStyleFootnote</code> as String	756
* 6.10.19 <code>NSFontTextStyleHeadline</code> as String	757
* 6.10.20 <code>NSFontTextStyleLargeTitle</code> as String	757
* 6.10.21 <code>NSFontTextStyleSubheadline</code> as String	757

	47
* 6.10.22 NSFontTextStyleTitle1 as String	757
* 6.10.23 NSFontTextStyleTitle2 as String	757
* 6.10.24 NSFontTextStyleTitle3 as String	757
* 6.10.26 Description as String	758
* 6.10.27 Handle as Integer	758

• 4 Cocoa	147
– 4.13.1 class NSMutableIndexSetMBS	285
* 4.13.3 Constructor	285
* 4.13.4 Constructor(index as Integer)	286
* 4.13.5 Constructor(indexes as NSMutableIndexSetMBS)	286
* 4.13.6 Constructor(StartIndex as Integer, Length as Integer)	287
* 4.13.7 containsIndex(index as Integer) as boolean	287
* 4.13.8 containsIndexes(indexes as NSMutableIndexSetMBS) as boolean	288
* 4.13.9 containsIndexesInRange(StartIndex as Integer, Length as Integer) as boolean	288
* 4.13.10 copy as NSMutableIndexSetMBS	288
* 4.13.11 count as Integer	289
* 4.13.12 countOfIndexesInRange(StartIndex as Integer, Length as Integer) as Integer	289
* 4.13.13 firstIndex as Integer	289
* 4.13.14 indexGreaterThanIndex(index as Integer) as Integer	290
* 4.13.15 indexGreaterThanOrEqualToIndex(index as Integer) as Integer	290
* 4.13.16 indexLessThanIndex(index as Integer) as Integer	290
* 4.13.17 indexLessThanOrEqualToIndex(index as Integer) as Integer	291
* 4.13.18 indexSet as NSMutableIndexSetMBS	291
* 4.13.19 indexSetWithIndex(index as Integer) as NSMutableIndexSetMBS	291
* 4.13.20 indexSetWithIndexesInRange(StartIndex as Integer, Length as Integer) as NSMutableIndexSetMBS	292
* 4.13.21 intersectsIndexesInRange(StartIndex as Integer, Length as Integer) as boolean	292
* 4.13.22 isEqualToIndexSet(other as NSMutableIndexSetMBS) as boolean	292
* 4.13.23 lastIndex as Integer	293
* 4.13.24 mutableCopy as NSMutableIndexSetMBS	293
* 4.13.25 Operator_Convert as string	294
* 4.13.26 Values as Integer()	294
* 4.13.28 Handle as Integer	294
– 4.14.1 class NSMutableInputStreamMBS	295
* 4.14.3 Constructor(filePath as string)	295
* 4.14.4 inputStreamWithData(data as MemoryBlock) as NSMutableInputStreamMBS	295
* 4.14.5 inputStreamWithFileAtPath(path as string) as NSMutableInputStreamMBS	295
* 4.14.6 inputStreamWithURL(URL as string) as NSMutableInputStreamMBS	296
* 4.14.7 LookAhead as MemoryBlock	296
* 4.14.8 Read(maxLength as Integer) as MemoryBlock	296
* 4.14.10 hasBytesAvailable as Boolean	296
– 4.15.1 class NSMutableKeyedArchiverMBS	297
* 4.15.3 archiverData as memoryblock	298
* 4.15.4 Constructor	298
* 4.15.5 Constructor(RequiringSecureCoding as Boolean)	298
* 4.15.6 finishEncoding	299

	49
* 4.15.8 outputFormat as Integer	299
– 4.16.1 class NSKeyedUnarchiverMBS	301
* 4.16.3 Constructor(data as memoryblock)	302
* 4.16.4 finishDecoding	302
– 4.17.1 class NSKeyValueObserverMBS	303
* 4.17.3 addObserver(keyPath as string, options as Integer = 5, context as Variant = nil)	303
* 4.17.4 Constructor(TargetHandle as Integer)	303
* 4.17.5 Destructor	304
* 4.17.6 NSKeyValueChangeIndexesKey as string	304
* 4.17.7 NSKeyValueChangeKindKey as string	304
* 4.17.8 NSKeyValueChangeNewKey as string	304
* 4.17.9 NSKeyValueChangeNotificationIsPriorKey as string	305
* 4.17.10 NSKeyValueChangeOldKey as string	305
* 4.17.11 removeObserver(keyPath as string, context as Variant = nil)	305
* 4.17.13 Handle as Integer	306
* 4.17.15 observedValueForKeyPathChanged(keyPath as string, target as Variant, change as dictionary, context as Variant, ChangeNSDictionaryRef as Integer) as boolean	306

• 12 Currency, Date and Time Format	1047
– 12.1.1 class NSLocaleDateMBS	1047
* 12.1.3 Constructor	1047
* 12.1.4 Constructor(locale as NSLocaleMBS)	1048
* 12.1.5 eraSymbols as string()	1048
* 12.1.6 longEraSymbols as string()	1048
* 12.1.7 monthSymbols as string()	1049
* 12.1.8 quarterSymbols as string()	1049
* 12.1.9 shortMonthSymbols as string()	1050
* 12.1.10 shortQuarterSymbols as string()	1050
* 12.1.11 shortStandaloneMonthSymbols as string()	1051
* 12.1.12 shortStandaloneQuarterSymbols as string()	1051
* 12.1.13 shortStandaloneWeekdaySymbols as string()	1052
* 12.1.14 shortWeekdaySymbols as string()	1052
* 12.1.15 standaloneMonthSymbols as string()	1053
* 12.1.16 standaloneQuarterSymbols as string()	1053
* 12.1.17 standaloneWeekdaySymbols as string()	1054
* 12.1.18 veryShortMonthSymbols as string()	1054
* 12.1.19 veryShortStandaloneMonthSymbols as string()	1054
* 12.1.20 veryShortStandaloneWeekdaySymbols as string()	1055
* 12.1.21 veryShortWeekdaySymbols as string()	1055
* 12.1.22 weekdaySymbols as string()	1056
* 12.1.24 AMSymbol as String	1056
* 12.1.25 dateFormat as String	1057
* 12.1.26 PMSymbol as String	1057
– 12.2.1 class NSLocaleMBS	1058
* 12.2.3 autoupdatingCurrentLocale as NSLocaleMBS	1058
* 12.2.4 availableLocaleIdentifiers as string()	1059
* 12.2.5 canonicalLanguageIdentifierFromString(s as string) as string	1059
* 12.2.6 canonicalLocaleIdentifierFromString(s as string) as string	1060
* 12.2.7 characterDirectionForLanguage(isoLangCode as string) as Integer	1060
* 12.2.8 commonISOCurrencyCodes as string()	1060
* 12.2.9 Constructor	1061
* 12.2.10 Constructor(Identifier as string)	1061
* 12.2.11 currentLocale as NSLocaleMBS	1062
* 12.2.12 displayName(key as string, value as string) as string	1062
* 12.2.13 ExemplarCharacterSet as Variant	1063
* 12.2.14 GetString(key as string) as string	1064
* 12.2.15 ISOCountryCodes as string()	1064
* 12.2.16 ISOCurrencyCodes as string()	1064
* 12.2.17 ISOLanguageCodes as string()	1065

* 12.2.18 lineDirectionForLanguage(isoLangCode as string) as Integer	1066
* 12.2.19 localeIdentifier as string	1067
* 12.2.20 localeIdentifierFromWindowsLocaleCode(code as Integer) as string	1067
* 12.2.21 NSBuddhistCalendar as string	1067
* 12.2.22 NSChineseCalendar as string	1067
* 12.2.23 NSGregorianCalendar as string	1068
* 12.2.24 NSHebrewCalendar as string	1068
* 12.2.25 NSIndianCalendar as string	1068
* 12.2.26 NSIslamicCalendar as string	1068
* 12.2.27 NSIslamicCivilCalendar as string	1068
* 12.2.28 NSISO8601Calendar as string	1069
* 12.2.29 NSJapaneseCalendar as string	1069
* 12.2.30 NSLocaleAlternateQuotationBeginDelimiterKey as string	1069
* 12.2.31 NSLocaleAlternateQuotationEndDelimiterKey as string	1069
* 12.2.32 NSLocaleCalendar as string	1070
* 12.2.33 NSLocaleCollationIdentifier as string	1070
* 12.2.34 NSLocaleCollatorIdentifier as string	1070
* 12.2.35 NSLocaleCountryCode as string	1070
* 12.2.36 NSLocaleCurrencyCode as string	1070
* 12.2.37 NSLocaleCurrencySymbol as string	1071
* 12.2.38 NSLocaleDecimalSeparator as string	1071
* 12.2.39 NSLocaleExemplarCharacterSet as string	1071
* 12.2.40 NSLocaleGroupingSeparator as string	1071
* 12.2.41 NSLocaleIdentifier as string	1071
* 12.2.42 NSLocaleLanguageCode as string	1072
* 12.2.43 NSLocaleMeasurementSystem as string	1072
* 12.2.44 NSLocaleQuotationBeginDelimiterKey as string	1072
* 12.2.45 NSLocaleQuotationEndDelimiterKey as string	1072
* 12.2.46 NSLocaleScriptCode as string	1072
* 12.2.47 NSLocaleUsesMetricSystem as string	1072
* 12.2.48 NSLocaleVariantCode as string	1073
* 12.2.49 NSPersianCalendar as string	1073
* 12.2.50 NSRepublicOfChinaCalendar as string	1073
* 12.2.51 preferredLanguages as string()	1073
* 12.2.52 systemLocale as NSLocaleMBS	1074
* 12.2.53 windowsLocaleCodeFromLocaleIdentifier(s as string) as Integer	1074
* 12.2.55 CollationIdentifier as String	1074
* 12.2.56 CountryCode as String	1075
* 12.2.57 CurrencyCode as String	1075
* 12.2.58 CurrencySymbol as String	1075
* 12.2.59 DateFull as NSLocaleDateMBS	1076
* 12.2.60 DateLong as NSLocaleDateMBS	1076

* 12.2.61	DateMedium as NSLocaleDateMBS	1076
* 12.2.62	DateShort as NSLocaleDateMBS	1076
* 12.2.63	DecimalSeparator as String	1077
* 12.2.64	GroupingSeparator as String	1077
* 12.2.65	Identifier as String	1077
* 12.2.66	LanguageCode as String	1078
* 12.2.67	MeasurementSystem as String	1078
* 12.2.68	NumberCurrency as NSLocaleNumberMBS	1078
* 12.2.69	NumberDecimal as NSLocaleNumberMBS	1079
* 12.2.70	NumberPercent as NSLocaleNumberMBS	1079
* 12.2.71	NumberScientific as NSLocaleNumberMBS	1079
* 12.2.72	NumberSpellOut as NSLocaleNumberMBS	1079
* 12.2.73	ScriptCode as String	1079
* 12.2.74	UsesMetricSystem as Boolean	1080
* 12.2.75	VariantCode as String	1080
– 12.3.1	class NSLocaleNumberMBS	1082
* 12.3.3	Constructor	1082
* 12.3.4	Constructor(locale as NSLocaleMBS)	1082
* 12.3.6	alwaysShowsDecimalSeparator as Boolean	1083
* 12.3.7	currencyCode as String	1083
* 12.3.8	currencyDecimalSeparator as String	1083
* 12.3.9	currencyGroupingSeparator as String	1084
* 12.3.10	currencySymbol as String	1084
* 12.3.11	decimalSeparator as String	1084
* 12.3.12	exponentSymbol as String	1085
* 12.3.13	format as String	1085
* 12.3.14	groupingSeparator as String	1085
* 12.3.15	hasThousandSeparators as Boolean	1086
* 12.3.16	internationalCurrencySymbol as String	1086
* 12.3.17	Lenient as Boolean	1086
* 12.3.18	localizesFormat as Boolean	1086
* 12.3.19	minusSign as String	1087
* 12.3.20	negativeFormat as String	1087
* 12.3.21	negativeInfinitySymbol as String	1087
* 12.3.22	negativePrefix as String	1088
* 12.3.23	negativeSuffix as String	1088
* 12.3.24	nilSymbol as String	1088
* 12.3.25	notANumberSymbol as String	1089
* 12.3.26	paddingCharacter as String	1089
* 12.3.27	PartialStringValidationEnabled as Boolean	1090
* 12.3.28	percentSymbol as String	1090

* 12.3.29 perMillSymbol as String	1090
* 12.3.30 plusSign as String	1090
* 12.3.31 positiveFormat as String	1091
* 12.3.32 positiveInfinitySymbol as String	1091
* 12.3.33 positivePrefix as String	1092
* 12.3.34 positiveSuffix as String	1092
* 12.3.35 thousandSeparator as String	1092
* 12.3.36 usesGroupingSeparator as Boolean	1092
* 12.3.37 usesSignificantDigits as Boolean	1093
* 12.3.38 zeroSymbol as String	1093

• 7 Cocoa Menus	763
– 7.1.1 class NSMenuItemMBS	763
* 7.1.3 clearAction	764
* 7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem)	764
* 7.1.5 Constructor(Handle as Integer)	764
* 7.1.6 Constructor(MenuItem as MenuItem)	765
* 7.1.7 Constructor(Other as NSMenuItemMBS)	766
* 7.1.8 Constructor(title as string="", keyEquivalent as string="")	766
* 7.1.9 CreateMenuItem(title as string="", keyEquivalent as string="")	766
* 7.1.10 CreateSeparator	767
* 7.1.11 MenuItem(DesktopMenuItem as DesktopMenuItem) as NSMenuItemMBS	767
* 7.1.12 MenuItem(MenuItem as MenuItem) as NSMenuItemMBS	767
* 7.1.13 sectionHeaderWithTitle(Title as String) as NSMenuItemMBS	768
* 7.1.14 separatorItem as NSMenuItemMBS	768
* 7.1.15 setAction(target as NSResponderMBS, selectorName as string)	768
* 7.1.16 setTitleWithMnemonic(title as String)	769
* 7.1.18 ActionSelector as String	769
* 7.1.19 Alternate as boolean	769
* 7.1.20 attributedTitle as NSAttributedStringMBS	769
* 7.1.21 Enabled as boolean	770
* 7.1.22 Handle as Integer	770
* 7.1.23 hasSubmenu as boolean	770
* 7.1.24 Identifier as String	770
* 7.1.25 image as NSImageMBS	771
* 7.1.26 indentationLevel as Integer	771
* 7.1.27 isHidden as boolean	771
* 7.1.28 isHiddenOrHasHiddenAncestor as boolean	771
* 7.1.29 isHighlighted as boolean	772
* 7.1.30 isSectionHeader as Boolean	772
* 7.1.31 isSeparatorItem as boolean	772
* 7.1.32 keyEquivalent as String	772
* 7.1.33 keyEquivalentModifierMask as Integer	773
* 7.1.34 menu as NSMenuMBS	774
* 7.1.35 mixedStateImage as NSImageMBS	774
* 7.1.36 offStateImage as NSImageMBS	775
* 7.1.37 onStateImage as NSImageMBS	775
* 7.1.38 parentItem as NSMenuItemMBS	775
* 7.1.39 state as Integer	775
* 7.1.40 submenu as NSMenuMBS	776
* 7.1.41 tag as Integer	776
* 7.1.42 Title as String	776

* 7.1.43	toolTip as String	776
* 7.1.44	userKeyEquivalent as String	776
* 7.1.45	view as NSViewMBS	777
* 7.1.46	usesUserKeyEquivalent as boolean	777
* 7.1.48	Action	777
* 7.1.49	validateMenuItem(menuItem as NSMenuItemMBS) as boolean	777
– 7.2.1	class NSMenuMBS	779
* 7.2.3	addItem(m as NSMenuItemMBS)	779
* 7.2.4	cancelTracking	780
* 7.2.5	cancelTrackingWithoutAnimation	780
* 7.2.6	Constructor(Handle as Integer)	780
* 7.2.7	Constructor(title as string=’’)	780
* 7.2.8	helpMenu as NSMenuMBS	781
* 7.2.9	indexOfItem(item as NSMenuItemMBS) as Integer	781
* 7.2.10	indexOfItemWithSelector(selector as string) as Integer	781
* 7.2.11	indexOfItemWithSubmenu(item as NSMenuMBS) as Integer	781
* 7.2.12	indexOfItemWithTag(tag as Integer) as Integer	781
* 7.2.13	indexOfItemWithTitle(title as string) as Integer	782
* 7.2.14	insertItem(m as NSMenuItemMBS, index as Integer)	782
* 7.2.15	Item(index as Integer) as NSMenuItemMBS	782
* 7.2.16	itemWithSelector(selector as string) as NSMenuItemMBS	782
* 7.2.17	mainMenu as NSMenuMBS	782
* 7.2.18	menuBarVisible as boolean	783
* 7.2.19	NSMenuDidAddItemNotification as string	783
* 7.2.20	NSMenuDidBeginTrackingNotification as string	783
* 7.2.21	NSMenuDidChangeItemNotification as string	783
* 7.2.22	NSMenuDidEndTrackingNotification as string	784
* 7.2.23	NSMenuDidRemoveItemNotification as string	784
* 7.2.24	NSMenuDidSendActionNotification as string	785
* 7.2.25	NSMenuWillSendActionNotification as string	785
* 7.2.26	paletteMenuWithColors(colors() as NSColorMBS, itemTitles() as String, templateImage as NSImageMBS = nil) as NSMenuMBS	785
* 7.2.27	performActionForItemAtIndex(index as Integer)	786
* 7.2.28	popupContextMenu(menu as NSMenuMBS, theEvent as NSEventMBS, view as NSViewMBS, font as NSFontMBS = nil)	787
* 7.2.29	popupMenuPositioningItem(item as NSMenuItemMBS, location as NSPointMBS, view as NSViewMBS = nil) as boolean	787
* 7.2.30	removeAllItems	788
* 7.2.31	removeItem(m as NSMenuItemMBS)	789
* 7.2.32	removeItemAtIndex(index as Integer)	789
* 7.2.33	selectedItems as NSMenuItemMBS()	789
* 7.2.34	setMenuBarVisible(value as boolean)	790

* 7.2.35 setSelectedItem(items() as NSMenuItemMBS)	790
* 7.2.36 update	790
* 7.2.37 windowsMenu as NSMenuMBS	790
* 7.2.39 allowsContextMenuPlugIns as boolean	790
* 7.2.40 autoenablesItems as Boolean	791
* 7.2.41 Font as NSFontMBS	791
* 7.2.42 Handle as Integer	791
* 7.2.43 highlightedItem as NSMenuItemMBS	791
* 7.2.44 Identifier as String	792
* 7.2.45 menuBarHeight as Double	792
* 7.2.46 minimumWidth as Double	792
* 7.2.47 numberOfItems as Integer	793
* 7.2.48 presentationStyle as Integer	793
* 7.2.49 selectionMode as Integer	793
* 7.2.50 showsStateColumn as boolean	793
* 7.2.51 size as NSSizeMBS	794
* 7.2.52 supermenu as NSMenuMBS	794
* 7.2.53 Title as String	794
* 7.2.54 userInterfaceLayoutDirection as Integer	794
* 7.2.56 DidClose	795
* 7.2.57 EnableMenuItems	795
* 7.2.58 PaletteSelectionChange	795
* 7.2.59 willHighlightItem(item as NSMenuItemMBS)	795
* 7.2.60 WillOpen	795

	57
• 9 Cocoa Text	925
– 9.3.1 class NSMutableAttributedStringMBS	1001
* 9.3.3 addAttribute(name as string, value as Variant, range as NSRangeMBS)	1001
* 9.3.4 addAttributes(attrs as Dictionary, range as NSRangeMBS)	1002
* 9.3.5 appendAttributedString(attrString as NSAttributedStringMBS)	1003
* 9.3.6 appendString(attrString as String)	1003
* 9.3.7 applyFontTraits(FontTraitMask as Integer, offset as Integer, length as Integer)	1004
* 9.3.8 AsCFMutableAttributedString as Variant	1004
* 9.3.9 beginEditing	1004
* 9.3.10 Constructor	1005
* 9.3.11 containsAttachmentsInRange(offset as Integer, length as Integer) as Boolean	1005
* 9.3.12 deleteCharactersInRange(range as NSRangeMBS)	1005
* 9.3.13 endEditing	1005
* 9.3.14 fixAttachmentAttributeInRange(offset as Integer, length as Integer)	1005
* 9.3.15 fixAttributesInRange(offset as Integer, length as Integer)	1006
* 9.3.16 fixFontAttributeInRange(offset as Integer, length as Integer)	1006
* 9.3.17 fixParagraphStyleAttributeInRange(offset as Integer, length as Integer)	1006
* 9.3.18 fontAttributesInRange(offset as Integer, length as Integer) as Dictionary	1006
* 9.3.19 insertAttributedString(attrString as NSAttributedStringMBS, location as UInt64)	1007
* 9.3.20 insertString(attrString as String, location as UInt64)	1007
* 9.3.21 removeAttribute(name as string, range as NSRangeMBS)	1007
* 9.3.22 replaceCharacters(search as String, text as String, Options as Integer = 1)	1007
* 9.3.23 replaceCharactersInRange(range as NSRangeMBS, attrString as NSAttributedStringMBS)	1008
* 9.3.24 replaceCharactersInRange(range as NSRangeMBS, text as string)	1008
* 9.3.25 rulerAttributesInRange(offset as Integer, length as Integer) as Dictionary	1009
* 9.3.26 setAlignment(alignment as Integer, offset as Integer, length as Integer)	1009
* 9.3.27 setAttributedString(attrString as NSAttributedStringMBS)	1009
* 9.3.28 setAttributes(attrs as Dictionary, range as NSRangeMBS)	1009
* 9.3.29 setBaseWritingDirection(writingDirection as Integer, offset as Integer, length as Integer)	1010
* 9.3.30 setString(attrString as String)	1010
* 9.3.31 subscriptRange(offset as Integer, length as Integer)	1010
* 9.3.32 superscriptRange(offset as Integer, length as Integer)	1011
* 9.3.33 unscriptRange(offset as Integer, length as Integer)	1011
* 9.3.34 updateAttachmentsFromPath(file as folderitem)	1011
* 9.3.35 updateAttachmentsFromPath(path as string)	1011

• 4 Cocoa	147
– 4.18.1 class NSMutableCharacterSetMBS	308
* 4.18.3 addCharactersInRange(aRange as NSRangeMBS)	308
* 4.18.4 addCharactersInString(aString as string)	309
* 4.18.5 Constructor	309
* 4.18.6 formIntersectionWithCharacterSet(otherset as NSMutableCharacterSetMBS)	309
* 4.18.7 formUnionWithCharacterSet(otherset as NSMutableCharacterSetMBS)	309
* 4.18.8 invert	309
* 4.18.9 removeCharactersInRange(aRange as NSRangeMBS)	310
* 4.18.10 removeCharactersInString(aString as string)	310
– 4.19.1 class NSMutableIndexSetMBS	311
* 4.19.3 addIndex(index as Integer)	311
* 4.19.4 addIndexes(indexes as NSIndexSetMBS)	311
* 4.19.5 addIndexesInRange(StartIndex as Integer, Length as Integer)	312
* 4.19.6 Constructor	312
* 4.19.7 Constructor(index as Integer)	312
* 4.19.8 Constructor(indexes as NSIndexSetMBS)	313
* 4.19.9 Constructor(StartIndex as Integer, Length as Integer)	313
* 4.19.10 removeAllIndexes	314
* 4.19.11 removeIndex(index as Integer)	314
* 4.19.12 removeIndexes(indexes as NSIndexSetMBS)	314
* 4.19.13 removeIndexesInRange(StartIndex as Integer, Length as Integer)	315
* 4.19.14 shiftIndexes(StartingAtIndex as Integer, delta as Integer)	315
– 4.20.1 class NSMutableURLRequestMBS	316
* 4.20.3 addValue(value as string, field as string)	316
* 4.20.4 Constructor(url as string)	317
* 4.20.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double)	317
* 4.20.6 setAllHTTPHeaderFields(headerFields as Dictionary)	317
* 4.20.7 setAttribution(Attribution as Integer)	318
* 4.20.8 setCachePolicy(policy as Integer)	318
* 4.20.9 setHTTPBody(data as MemoryBlock)	318
* 4.20.10 setHTTPMethod(HTTPMethod as string)	318
* 4.20.11 setHTTPShouldHandleCookies(should as boolean)	319
* 4.20.12 setHTTPShouldUsePipelining(shouldUsePipelining as boolean)	319
* 4.20.13 setMainDocumentURL(url as string)	319
* 4.20.14 setNetworkServiceType(networkServiceType as Integer)	320
* 4.20.15 setTimeoutInterval(seconds as Double)	320
* 4.20.16 setURL(url as string)	320
* 4.20.17 setValue(value as string, field as string)	321

• 14 Notifications	1097
– 14.2.1 class NSNotificationCenterMBS	1102
* 14.2.3 addObserver(observer as NSNotificationCenterObserverMBS, name as string="", theObject as Variant=nil)	1102
* 14.2.4 Constructor	1102
* 14.2.5 defaultCenter as NSNotificationCenterMBS	1103
* 14.2.6 postNotification(notification as NSNotificationCenterMBS)	1103
* 14.2.7 postNotificationName(name as string)	1103
* 14.2.8 postNotificationName(name as string, theObject as Variant)	1103
* 14.2.9 postNotificationName(name as string, theObject as Variant, userInfo as dictionary)	1103
* 14.2.10 removeObserver(observer as NSNotificationCenterObserverMBS)	1104
* 14.2.11 removeObserver(observer as NSNotificationCenterObserverMBS, name as string, theObject as Variant=nil)	1104
* 14.2.13 Handle as Integer	1105
– 14.3.1 class NSNotificationMBS	1106
* 14.3.3 Constructor(handle as Integer)	1106
* 14.3.4 Constructor(name as string, theObject as Variant = nil, userInfo as dictionary = nil)	1107
* 14.3.5 notificationWithName(name as string, theObject as Variant = nil, userInfo as dictionary = nil) as NSNotificationMBS	1107
* 14.3.6 Print	1107
* 14.3.8 description as string	1108
* 14.3.9 Handle as Integer	1108
* 14.3.10 name as string	1108
* 14.3.11 objectHandle as Integer	1108
* 14.3.12 objectVariant as Variant	1109
* 14.3.13 userInfo as dictionary	1109
– 14.4.1 class NSNotificationCenterObserverMBS	1111
* 14.4.3 Constructor	1111
* 14.4.4 Destructor	1111
* 14.4.6 Handle as Integer	1112
* 14.4.8 GotNotification(notification as NSNotificationCenterMBS)	1112

• 4 Cocoa	147
– 4.21.1 class NSOutputStreamMBS	322
* 4.21.3 Constructor	322
* 4.21.4 Constructor(filePath as string, append as boolean)	322
* 4.21.5 OutputData as MemoryBlock	323
* 4.21.6 outputStreamToFileAtPath(filePath as string, append as boolean) as NSOutputStreamMBS	323
* 4.21.7 outputStreamToMemory as NSOutputStreamMBS	323
* 4.21.8 outputStreamWithURL(fileURL as string, append as boolean) as NSOutputStreamMBS	323
* 4.21.9 write(data as MemoryBlock) as Integer	324
* 4.21.11 hasSpaceAvailable as Boolean	324
– 4.22.1 class NSPanelMBS	325
* 4.22.3 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer, deferCreation as boolean)	325
* 4.22.4 RunAlertPanel(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string) as Integer	326
* 4.22.5 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer	327
* 4.22.6 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer	327
* 4.22.7 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer	327
* 4.22.8 RunCriticalAlertPanel(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string) as Integer	328
* 4.22.9 RunCriticalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer	328
* 4.22.10 RunCriticalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer	328
* 4.22.11 RunCriticalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer	328
* 4.22.12 RunInformationalAlertPanel(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string) as Integer	329
* 4.22.13 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer	329
* 4.22.14 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer	329

- * 4.22.15 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 330
- * 4.22.17 becomesKeyOnlyIfNeeded as boolean 330
- * 4.22.18 isFloatingPanel as boolean 330
- * 4.22.19 worksWhenModal as boolean 331

• 3 Clipboard	121
– 3.1.1 class NSPasteboardItemDataProviderMBS	121
* 3.1.3 Constructor	121
* 3.1.4 Destructor	121
* 3.1.6 Handle as Integer	122
* 3.1.8 Finished(Pasteboard as NSPasteboardMBS)	122
* 3.1.9 provideDataForType(Pasteboard as NSPasteboardMBS, item as NSPasteboardItemMBS, type as string)	122
– 3.2.1 class NSPasteboardItemMBS	123
* 3.2.3 availableTypeFromArray(types() as string) as string	124
* 3.2.4 Constructor	124
* 3.2.5 Destructor	124
* 3.2.6 setDataProviderForType(dataProvider as NSPasteboardItemDataProviderMBS, types() as string) as boolean	124
* 3.2.7 types as string()	125
* 3.2.9 dataProvider as NSPasteboardItemDataProviderMBS	125
* 3.2.10 Handle as Integer	125
* 3.2.11 dataForType(type as string) as memoryblock	125
* 3.2.12 propertyListForType(type as string) as Variant	126
* 3.2.13 stringForType(type as string) as string	126
– 3.3.1 class NSPasteboardMBS	127
* 3.3.3 addType(type as string) as Integer	127
* 3.3.4 addTypes(types() as string) as Integer	128
* 3.3.5 changeCount as Integer	129
* 3.3.6 clearContents as Integer	129
* 3.3.7 Constructor	129
* 3.3.8 Constructor(name as string)	130
* 3.3.9 declareType(type as string) as Integer	130
* 3.3.10 declareTypes(types() as string) as Integer	131
* 3.3.11 generalPasteboard as NSPasteboardMBS	131
* 3.3.12 name as string	131
* 3.3.13 NSColorPboardType as string	132
* 3.3.14 NSDragPboard as string	132
* 3.3.15 NSFileNamesPboardType as string	132
* 3.3.16 NSFilesPromisePboardType as string	132
* 3.3.17 NSFindPboard as string	133
* 3.3.18 NSFontPboard as string	133
* 3.3.19 NSFontPboardType as string	133
* 3.3.20 NSGeneralPboard as string	134
* 3.3.21 NSHTMLPboardType as string	134
* 3.3.22 NSInkTextPboardType as string	134

* 3.3.23 NSMultipleTextSelectionPboardType as string	134
* 3.3.24 NSPasteboardTypeColor as string	135
* 3.3.25 NSPasteboardTypeFindPanelSearchOptions as string	135
* 3.3.26 NSPasteboardTypeFont as string	135
* 3.3.27 NSPasteboardTypeHTML as string	135
* 3.3.28 NSPasteboardTypeMultipleTextSelection as string	135
* 3.3.29 NSPasteboardTypePDF as string	136
* 3.3.30 NSPasteboardTypePNG as string	136
* 3.3.31 NSPasteboardTypeRTF as string	136
* 3.3.32 NSPasteboardTypeRTFD as string	136
* 3.3.33 NSPasteboardTypeRuler as string	136
* 3.3.34 NSPasteboardTypeSound as string	137
* 3.3.35 NSPasteboardTypeString as string	137
* 3.3.36 NSPasteboardTypeTabularText as string	137
* 3.3.37 NSPasteboardTypeTIFF as string	137
* 3.3.38 NSPDFPboardType as string	137
* 3.3.39 NSICTPboardType as string	138
* 3.3.40 NSPostScriptPboardType as string	138
* 3.3.41 NSRTFDPboardType as string	138
* 3.3.42 NSRTFPboardType as string	138
* 3.3.43 NSRulerPboard as string	139
* 3.3.44 NSRulerPboardType as string	139
* 3.3.45 NSSstringPboardType as string	139
* 3.3.46 NSTabularTextPboardType as string	139
* 3.3.47 NSTIFFPboardType as string	139
* 3.3.48 NSURLPboardType as string	140
* 3.3.49 NSVCardPboardType as string	140
* 3.3.50 pasteboardItems as NSPasteboardItemMBS()	140
* 3.3.51 pasteboardWithName(name as string) as NSPasteboardMBS	140
* 3.3.52 pasteboardWithUniqueName as NSPasteboardMBS	141
* 3.3.53 releaseGlobally	141
* 3.3.54 SetPasteboardItems(items() as NSPasteboardItemMBS) as Boolean	141
* 3.3.55 SetPasteboardItems(items() as Variant) as boolean	141
* 3.3.56 types as string()	142
* 3.3.57 URLFromPasteboard as string	143
* 3.3.58 writeURLToPasteboard(URL as string)	143
* 3.3.60 Handle as Integer	143
* 3.3.61 dataForType(type as string) as Memoryblock	143
* 3.3.62 propertyListForType(type as string) as Variant	144
* 3.3.63 stringForType(type as string) as string	144

• 6 Cocoa Drawing	551
– 6.11.1 class NSPDFImageRepMBS	759
* 6.11.3 Constructor(data as Memoryblock)	759
* 6.11.4 imageRepWithData(data as Memoryblock) as NSPDFImageRepMBS	759
* 6.11.5 PDFRepresentation as Memoryblock	759
* 6.11.7 bounds as NSRectMBS	760
* 6.11.8 currentPage as Integer	760
* 6.11.9 pageCount as Integer	760
– 6.12.1 class NSPICTImageRepMBS	761
* 6.12.3 Constructor(data as Memoryblock)	761
* 6.12.4 imageRepWithData(data as Memoryblock) as NSPICTImageRepMBS	761
* 6.12.5 PICTRepresentation as Memoryblock	761
* 6.12.7 boundingBox as NSRectMBS	762

• 15 Process	1113
– 15.1.1 class NSProcessInfoActivityMBS	1113
* 15.1.3 Constructor	1114
* 15.1.4 Destructor	1114
* 15.1.6 Handle as Integer	1114
* 15.1.7 Options as Integer	1114
* 15.1.8 Reason as String	1114
– 15.2.1 class NSProcessInfoMBS	1115
* 15.2.3 argument(index as Integer) as string	1116
* 15.2.4 arguments as string()	1116
* 15.2.5 beginActivity(options as Integer, reason as string) as NSProcessInfoActivityMBS	1116
* 15.2.6 Constructor	1117
* 15.2.7 disableAutomaticTermination(Reason as string)	1117
* 15.2.8 disableSuddenTermination	1117
* 15.2.9 enableAutomaticTermination(Reason as string)	1118
* 15.2.10 enableSuddenTermination	1118
* 15.2.11 endActivity(activity as NSProcessInfoActivityMBS)	1118
* 15.2.12 NSActivityLatencyCritical as UInt64	1119
* 15.2.13 NSProcessInfoPowerStateDidChangeNotification as String	1119
* 15.2.14 NSProcessInfoThermalStateDidChangeNotification as String	1119
* 15.2.15 operationSystemVersion(byref Major as Integer, byref Minor as Integer, byref Patch as Integer)	1120
* 15.2.16 processInfo as NSProcessInfoMBS	1120
* 15.2.18 activeProcessorCount as Integer	1120
* 15.2.19 argumentsCount as Integer	1120
* 15.2.20 automaticTerminationSupportEnabled as boolean	1121
* 15.2.21 environment as dictionary	1121
* 15.2.22 globallyUniqueString as string	1121
* 15.2.23 Handle as Integer	1122
* 15.2.24 hostName as string	1122
* 15.2.25 isiOSAppOnMac as Boolean	1122
* 15.2.26 isLowPowerModeEnabled as Boolean	1123
* 15.2.27 operatingSystem as Integer	1123
* 15.2.28 operatingSystemName as string	1123
* 15.2.29 operatingSystemVersionString as string	1124
* 15.2.30 operationSystemVersionMajor as Integer	1124
* 15.2.31 operationSystemVersionMinor as Integer	1124
* 15.2.32 operationSystemVersionPatch as Integer	1124
* 15.2.33 physicalMemory as UInt64	1124
* 15.2.34 processIdentifier as Integer	1125
* 15.2.35 processName as string	1125
* 15.2.36 processorCount as Integer	1126
* 15.2.37 systemUptime as Double	1126
* 15.2.38 thermalState as Integer	1126

• 4 Cocoa	147
– 4.23.1 class NSResponderMBS	333
* 4.23.3 beginGestureWithEvent(e as NSEventMBS)	333
* 4.23.4 cancelOperation	333
* 4.23.5 capitalizeWord	334
* 4.23.6 centerSelectionInVisibleArea	334
* 4.23.7 changeCaseOfLetter	334
* 4.23.8 complete	334
* 4.23.9 Constructor	334
* 4.23.10 cursorUpdate(e as NSEventMBS)	335
* 4.23.11 deleteBackward	335
* 4.23.12 deleteBackwardByDecomposingPreviousCharacter	335
* 4.23.13 deleteForward	335
* 4.23.14 deleteToBeginningOfLine	335
* 4.23.15 deleteToBeginningOfParagraph	336
* 4.23.16 deleteToEndOfLine	336
* 4.23.17 deleteToEndOfParagraph	336
* 4.23.18 deleteToMark	336
* 4.23.19 deleteWordBackward	337
* 4.23.20 deleteWordForward	337
* 4.23.21 endGestureWithEvent(e as NSEventMBS)	337
* 4.23.22 flagsChanged(e as NSEventMBS)	337
* 4.23.23 flushBufferedKeyEvents	337
* 4.23.24 helpRequested(e as NSEventMBS)	337
* 4.23.25 indent	338
* 4.23.26 insertBacktab	338
* 4.23.27 insertContainerBreak	338
* 4.23.28 insertDoubleQuoteIgnoringSubstitution	338
* 4.23.29 insertLineBreak	339
* 4.23.30 insertNewline	339
* 4.23.31 insertNewlineIgnoringFieldEditor	339
* 4.23.32 insertParagraphSeparator	339
* 4.23.33 insertSingleQuoteIgnoringSubstitution	339
* 4.23.34 insertTab	340
* 4.23.35 insertTabIgnoringFieldEditor	340
* 4.23.36 keyDown(e as NSEventMBS)	340
* 4.23.37 keyUp(e as NSEventMBS)	340
* 4.23.38 lowercaseWord	340
* 4.23.39 magnifyWithEvent(e as NSEventMBS)	341
* 4.23.40 makeBaseWritingDirectionLeftToRight	341
* 4.23.41 makeBaseWritingDirectionNatural	341

* 4.23.42	makeBaseWritingDirectionRightToLeft	342
* 4.23.43	makeTextWritingDirectionLeftToRight	342
* 4.23.44	makeTextWritingDirectionNatural	343
* 4.23.45	makeTextWritingDirectionRightToLeft	343
* 4.23.46	mouseDown(e as NSEventMBS)	343
* 4.23.47	mouseDragged(e as NSEventMBS)	344
* 4.23.48	mouseEntered(e as NSEventMBS)	344
* 4.23.49	mouseExited(e as NSEventMBS)	344
* 4.23.50	mouseMoved(e as NSEventMBS)	344
* 4.23.51	mouseUp(e as NSEventMBS)	344
* 4.23.52	moveBackward	344
* 4.23.53	moveBackwardAndModifySelection	345
* 4.23.54	moveDown	345
* 4.23.55	moveDownAndModifySelection	345
* 4.23.56	moveForward	346
* 4.23.57	moveForwardAndModifySelection	346
* 4.23.58	moveLeft	346
* 4.23.59	moveLeftAndModifySelection	346
* 4.23.60	moveParagraphBackwardAndModifySelection	347
* 4.23.61	moveParagraphForwardAndModifySelection	347
* 4.23.62	moveRight	347
* 4.23.63	moveRightAndModifySelection	348
* 4.23.64	moveToBeginningOfDocument	348
* 4.23.65	moveToBeginningOfDocumentAndModifySelection	348
* 4.23.66	moveToBeginningOfLine	348
* 4.23.67	moveToBeginningOfLineAndModifySelection	349
* 4.23.68	moveToBeginningOfParagraph	349
* 4.23.69	moveToBeginningOfParagraphAndModifySelection	349
* 4.23.70	moveToEndOfDocument	349
* 4.23.71	moveToEndOfDocumentAndModifySelection	349
* 4.23.72	moveToEndOfLine	350
* 4.23.73	moveToEndOfLineAndModifySelection	350
* 4.23.74	moveToEndOfParagraph	350
* 4.23.75	moveToEndOfParagraphAndModifySelection	350
* 4.23.76	moveToLeftEndOfLine	350
* 4.23.77	moveToLeftEndOfLineAndModifySelection	351
* 4.23.78	moveToRightEndOfLine	351
* 4.23.79	moveToRightEndOfLineAndModifySelection	351
* 4.23.80	moveUp	351
* 4.23.81	moveUpAndModifySelection	351
* 4.23.82	moveWordBackward	352
* 4.23.83	moveWordBackwardAndModifySelection	352

* 4.23.84	moveWordForward	352
* 4.23.85	moveWordForwardAndModifySelection	352
* 4.23.86	moveWordLeft	353
* 4.23.87	moveWordLeftAndModifySelection	353
* 4.23.88	moveWordRight	353
* 4.23.89	moveWordRightAndModifySelection	354
* 4.23.90	otherMouseDown(e as NSEventMBS)	354
* 4.23.91	otherMouseDragged(e as NSEventMBS)	354
* 4.23.92	otherMouseUp(e as NSEventMBS)	355
* 4.23.93	pageDown	355
* 4.23.94	pageDownAndModifySelection	355
* 4.23.95	pageUp	355
* 4.23.96	pageUpAndModifySelection	355
* 4.23.97	performMnemonic(theString as string) as boolean	356
* 4.23.98	presentError(e as NSErrorMBS) as boolean	356
* 4.23.99	rightMouseDown(e as NSEventMBS)	356
* 4.23.100	rightMouseDragged(e as NSEventMBS)	357
* 4.23.101	rightMouseUp(e as NSEventMBS)	357
* 4.23.102	rotateWithEvent(e as NSEventMBS)	357
* 4.23.103	scrollLineDown	358
* 4.23.104	scrollLineUp	358
* 4.23.105	scrollPageDown	358
* 4.23.106	scrollPageUp	358
* 4.23.107	scrollToBeginningOfDocument	358
* 4.23.108	scrollToEndOfDocument	359
* 4.23.109	scrollWheel(e as NSEventMBS)	359
* 4.23.110	selectAll	359
* 4.23.111	selectLine	359
* 4.23.112	selectParagraph	359
* 4.23.113	selectToMark	360
* 4.23.114	selectWord	360
* 4.23.115	setMark	360
* 4.23.116	showContextHelp	360
* 4.23.117	swapWithMark	360
* 4.23.118	swipeWithEvent(e as NSEventMBS)	361
* 4.23.119	tabletPoint(e as NSEventMBS)	361
* 4.23.120	tabletProximity(e as NSEventMBS)	361
* 4.23.121	transpose	361
* 4.23.122	transposeWords	362
* 4.23.123	undoManager as NSUndoManagerMBS	362
* 4.23.124	uppercaseWord	362
* 4.23.125	yank	362

	69
* 4.23.127 Handle as Integer	362
* 4.23.128 menu as NSMenuMBS	363
* 4.23.129 nextResponder as NSResponderMBS	363
– 4.24.1 class NSScreenMBS	364
* 4.24.3 backingAlignedRect(r as NSRectMBS, options as UInt64) as NSRectMBS	364
* 4.24.4 Constructor	365
* 4.24.5 convertRectFromBacking(r as NSRectMBS) as NSRectMBS	365
* 4.24.6 convertRectToBacking(r as NSRectMBS) as NSRectMBS	365
* 4.24.7 NSScreenColorSpaceDidChangeNotification as string	365
* 4.24.8 screens as NSScreenMBS()	366
* 4.24.9 supportedWindowDepths as Integer()	366
* 4.24.11 backingScaleFactor as Double	366
* 4.24.12 colorSpace as Variant	367
* 4.24.13 deepestScreen as NSScreenMBS	367
* 4.24.14 depth as Integer	367
* 4.24.15 deviceDescription as dictionary	368
* 4.24.16 firstScreen as NSScreenMBS	368
* 4.24.17 frame as NSRectMBS	368
* 4.24.18 Handle as Integer	369
* 4.24.19 localizedName as String	369
* 4.24.20 mainScreen as NSScreenMBS	369
* 4.24.21 screensHaveSeparateSpaces as Boolean	370
* 4.24.22 secondScreen as NSScreenMBS	370
* 4.24.23 userSpaceScaleFactor as Double	370
* 4.24.24 visibleFrame as NSRectMBS	371

• 9 Cocoa Text	925
– 9.4.1 class NSShadowMBS	1012
* 9.4.3 Constructor	1012
* 9.4.4 copy as NSShadowMBS	1012
* 9.4.5 set	1013
* 9.4.7 Handle as Integer	1013
* 9.4.8 shadowBlurRadius as Double	1013
* 9.4.9 shadowColor as NSColorMBS	1013
* 9.4.10 shadowOffset as NSSizeMBS	1014

	71
• 4 Cocoa	147
– 4.25.1 class NSSoundDelegateMBS	372
* 4.25.3 SoundFinished(s as NSSoundMBS, didFinishPlaying as boolean)	372
– 4.26.1 class NSSoundMBS	373
* 4.26.3 availableSounds as string()	373
* 4.26.4 canInitWithPasteboard as boolean	374
* 4.26.5 channelMapping as Integer()	374
* 4.26.6 Constructor	374
* 4.26.7 Constructor(data as MemoryBlock)	375
* 4.26.8 Constructor(file as folderitem, ByReference as boolean)	375
* 4.26.9 Constructor(url as string, ByReference as boolean)	376
* 4.26.10 duration as Double	376
* 4.26.11 isPlaying as boolean	376
* 4.26.12 name as string	376
* 4.26.13 NSSoundPboardType as string	376
* 4.26.14 pause as boolean	377
* 4.26.15 play as boolean	377
* 4.26.16 resume as boolean	377
* 4.26.17 setChannelMapping(mapping() as Integer)	377
* 4.26.18 setDelegate(DelegateHandler as NSSoundDelegateMBS)	377
* 4.26.19 setName(name as string) as boolean	378
* 4.26.20 soundNamed(name as string) as NSSoundMBS	378
* 4.26.21 soundUnfilteredFileTypes as string()	379
* 4.26.22 soundUnfilteredPasteboardTypes as string()	379
* 4.26.23 soundUnfilteredTypes as string()	379
* 4.26.24 soundWithContentsOfFile(file as folderitem, ByReference as boolean) as NSSoundMBS	379
* 4.26.25 soundWithContentsOfURL(url as string, ByReference as boolean) as NSSoundMBS	380
* 4.26.26 soundWithData(data as MemoryBlock) as NSSoundMBS	380
* 4.26.27 soundWithPasteboard as NSSoundMBS	380
* 4.26.28 stop as boolean	381
* 4.26.29 writeToPasteboard	381
* 4.26.31 Handle as Integer	381
* 4.26.32 currentTime as Double	381
* 4.26.33 loops as boolean	381
* 4.26.34 playbackDeviceIdentifier as string	382
* 4.26.35 volume as Double	382
– 4.27.1 class NSStreamMBS	383
* 4.27.3 Close	383
* 4.27.4 Constructor	383

* 4.27.5 Open	383
* 4.27.6 SetPosition(pos as Int64) as boolean	384
* 4.27.8 Error as NSErrorMBS	384
* 4.27.9 Handle as Integer	384
* 4.27.10 position as Int64	384
* 4.27.11 Status as Integer	384

• 9 Cocoa Text	925
– 9.5.1 class NSTextAttachmentMBS	1015
* 9.5.3 attributedStringWithAttachment(attachment as NSTextAttachmentMBS) as NSAttributedStringMBS	1015
* 9.5.4 Constructor(fileWrapper as NSFileWrapperMBS)	1016
* 9.5.5 Constructor(image as NSImageMBS)	1016
* 9.5.7 attachmentCell as Variant	1017
* 9.5.8 fileWrapper as NSFileWrapperMBS	1017
* 9.5.9 Handle as Integer	1017
– 9.6.1 class NSTextBlockMBS	1019
* 9.6.3 borderColor(edge as Integer) as NSColorMBS	1019
* 9.6.4 Constructor	1019
* 9.6.5 copy as NSTextBlockMBS	1019
* 9.6.6 setBorderColor(color as NSColorMBS)	1020
* 9.6.7 setBorderColor(color as NSColorMBS, edge as Integer)	1020
* 9.6.8 setContentWidth(value as double, ValueType as Integer)	1020
* 9.6.9 setValue(value as double, ValueType as Integer, dimension as Integer)	1020
* 9.6.10 setWidth(value as double, ValueType as Integer, Layer as Integer)	1021
* 9.6.11 setWidth(value as double, ValueType as Integer, Layer as Integer, edge as Integer)	1021
* 9.6.12 valueForDimension(dimension as Integer) as Double	1021
* 9.6.13 valueTypeForDimension(dimension as Integer) as Integer	1021
* 9.6.14 width(layer as Integer, edge as Integer) as double	1022
* 9.6.15 widthValueType(layer as Integer, edge as Integer) as Integer	1022
* 9.6.17 backgroundColor as NSColorMBS	1022
* 9.6.18 contentWidth as Double	1022
* 9.6.19 contentWidthValueType as Integer	1023
* 9.6.20 Handle as Integer	1023
* 9.6.21 verticalAlignment as Integer	1023
– 9.7.1 class NSTextListMBS	1025
* 9.7.3 Constructor(format as String, OptionsMask as Integer = 0)	1025
* 9.7.4 copy as NSTextListMBS	1026
* 9.7.5 markerForItemNumber(ItemNum as Integer) as String	1027
* 9.7.6 NSTextListMarkerBox as String	1027
* 9.7.7 NSTextListMarkerCheck as String	1027
* 9.7.8 NSTextListMarkerCircle as String	1027
* 9.7.9 NSTextListMarkerDecimal as String	1027
* 9.7.10 NSTextListMarkerDiamond as String	1028
* 9.7.11 NSTextListMarkerDisc as String	1028
* 9.7.12 NSTextListMarkerHyphen as String	1028
* 9.7.13 NSTextListMarkerLowercaseAlpha as String	1028

* 9.7.14	<code>NSTextListMarkerLowercaseHexadecimal</code> as String	1028
* 9.7.15	<code>NSTextListMarkerLowercaseLatin</code> as String	1028
* 9.7.16	<code>NSTextListMarkerLowercaseRoman</code> as String	1029
* 9.7.17	<code>NSTextListMarkerOctal</code> as String	1029
* 9.7.18	<code>NSTextListMarkerSquare</code> as String	1029
* 9.7.19	<code>NSTextListMarkerUppercaseAlpha</code> as String	1029
* 9.7.20	<code>NSTextListMarkerUppercaseHexadecimal</code> as String	1029
* 9.7.21	<code>NSTextListMarkerUppercaseLatin</code> as String	1030
* 9.7.22	<code>NSTextListMarkerUppercaseRoman</code> as String	1030
* 9.7.24	<code>Handle</code> as Integer	1030
* 9.7.25	<code>listOptions</code> as Integer	1030
* 9.7.26	<code>markerFormat</code> as String	1030
* 9.7.27	<code>startingItemNumber</code> as Integer	1031
– 9.8.1	class <code>NSTextTableBlockMBS</code>	1032
* 9.8.3	Constructor	1032
* 9.8.4	Constructor(<code>table</code> as <code>NSTextTableMBS</code> , <code>startingRow</code> as Integer, <code>rowSpan</code> as Integer, <code>startingColumn</code> as Integer, <code>colSpan</code> as Integer)	1032
* 9.8.6	<code>columnSpan</code> as Integer	1033
* 9.8.7	<code>rowSpan</code> as Integer	1033
* 9.8.8	<code>startingColumn</code> as Integer	1033
* 9.8.9	<code>startingRow</code> as Integer	1033
* 9.8.10	<code>table</code> as <code>NSTextTableMBS</code>	1033
– 9.9.1	class <code>NSTextTableMBS</code>	1034
* 9.9.3	Constructor	1034
* 9.9.5	<code>collapsesBorders</code> as Boolean	1034
* 9.9.6	<code>hidesEmptyCells</code> as Boolean	1034
* 9.9.7	<code>layoutAlgorithm</code> as Integer	1035
* 9.9.8	<code>numberOfColumns</code> as Integer	1035

	75
• 4 Cocoa	147
– 4.28.1 class NSTimeZoneMBS	386
* 4.28.3 Constructor	386
* 4.28.4 Constructor(name as string)	387
* 4.28.5 copy as NSTimeZoneMBS	387
* 4.28.6 defaultTimeZone as NSTimeZoneMBS	387
* 4.28.7 isEqualToTimeZone(timeZone as NSTimeZoneMBS) as boolean	388
* 4.28.8 knownTimeZoneNames as string()	388
* 4.28.9 localTimeZone as NSTimeZoneMBS	388
* 4.28.10 Print	389
* 4.28.11 systemTimeZone as NSTimeZoneMBS	389
* 4.28.12 timeZoneForSecondsFromGMT(seconds as Integer) as NSTimeZoneMBS	390
* 4.28.13 timeZoneWithName(name as string) as NSTimeZoneMBS	390
* 4.28.15 abbreviation as string	390
* 4.28.16 DaylightSavingTimeOffset as Double	391
* 4.28.17 description as string	391
* 4.28.18 Handle as Integer	391
* 4.28.19 isDaylightSavingTime as Boolean	391
* 4.28.20 name as string	392
* 4.28.21 SecondsFromGMT as Double	392
* 4.28.22 abbreviationDictionary as Dictionary	392
– 4.29.1 class NSTouchMBS	393
* 4.29.3 Constructor	394
* 4.29.4 locationInView(View as NSViewMBS) as NSPointMBS	394
* 4.29.5 previousLocationInView(View as NSViewMBS) as NSPointMBS	394
* 4.29.7 deviceSize as NSSizeMBS	395
* 4.29.8 force as Double	395
* 4.29.9 Handle as Integer	395
* 4.29.10 identity as MemoryBlock	395
* 4.29.11 majorRadius as Double	396
* 4.29.12 majorRadiusTolerance as Double	396
* 4.29.13 maximumPossibleForce as Double	396
* 4.29.14 normalizedPosition as NSPointMBS	396
* 4.29.15 phase as Integer	397
* 4.29.16 resting as Boolean	397
* 4.29.17 tapCount as Integer	397
* 4.29.18 timestamp as Double	397
* 4.29.19 type as Integer	398
* 4.29.20 view as NSViewMBS	398
* 4.29.21 window as NSWindowMBS	398

• 9 Cocoa Text	925
– 9.10.1 class NSUndoManagerMBS	1036
* 9.10.3 beginUndoGrouping	1036
* 9.10.4 canRedo as boolean	1036
* 9.10.5 canUndo as boolean	1036
* 9.10.6 Constructor	1037
* 9.10.7 disableUndoRegistration	1037
* 9.10.8 enableUndoRegistration	1037
* 9.10.9 endUndoGrouping	1037
* 9.10.10 groupingLevel as Integer	1038
* 9.10.11 isRedoing as boolean	1038
* 9.10.12 isUndoing as boolean	1038
* 9.10.13 isUndoRegistrationEnabled as boolean	1038
* 9.10.14 redo	1038
* 9.10.15 redoActionName as string	1039
* 9.10.16 redoMenuItemTitle as string	1039
* 9.10.17 redoMenuItemTitleForUndoActionName(actionName as string) as string	1039
* 9.10.18 removeAllActions	1039
* 9.10.19 setActionName(actionName as string)	1039
* 9.10.20 undo	1040
* 9.10.21 undoActionName as string	1040
* 9.10.22 undoMenuItemTitle as string	1040
* 9.10.23 undoMenuItemTitleForUndoActionName(actionName as string) as string	1040
* 9.10.24 undoNestedGroup	1040
* 9.10.26 Handle as Integer	1041
* 9.10.27 groupsByEvent as boolean	1041
* 9.10.28 levelsOfUndo as Integer	1041

• 8 Cocoa Networking	797
– 8.2.1 class NSURLAuthenticationChallengeMBS	800
* 8.2.3 cancelAuthenticationChallenge	800
* 8.2.4 Constructor	800
* 8.2.5 continueWithoutCredentialForAuthenticationChallenge	800
* 8.2.6 error as NSErrorMBS	800
* 8.2.7 failureResponse as NSURLResponseMBS	801
* 8.2.8 previousFailureCount as Integer	801
* 8.2.9 proposedCredential as NSURLCredentialMBS	801
* 8.2.10 protectionSpace as NSURLProtectionSpaceMBS	801
* 8.2.11 useCredential(credential as NSURLCredentialMBS)	801
* 8.2.13 Handle as Integer	802
– 8.3.1 class NSURLCacheMBS	803
* 8.3.3 cachedResponseForRequest(request as NSURLRequestMBS) as NSCachedURLResponseMBS	803
* 8.3.4 Constructor(memoryCapacity as UInt64, diskCapacity as UInt64, diskPath as folderitem)	803
* 8.3.5 removeAllCachedResponses	804
* 8.3.6 removeCachedResponseForRequest(request as NSURLRequestMBS)	804
* 8.3.7 removeCachedResponsesSinceDate(d as date)	804
* 8.3.8 removeCachedResponsesSinceDate(d as dateTime)	804
* 8.3.9 setSharedURLCache(cache as NSURLCacheMBS)	805
* 8.3.10 sharedURLCache as NSURLCacheMBS	805
* 8.3.11 storeCachedResponse(cachedResponse as NSCachedURLResponseMBS, request as NSURLRequestMBS)	805
* 8.3.13 currentDiskUsage as UInt64	806
* 8.3.14 currentMemoryUsage as UInt64	806
* 8.3.15 diskCapacity as UInt64	806
* 8.3.16 Handle as Integer	806
* 8.3.17 memoryCapacity as UInt64	806
– 8.4.1 class NSURLConnectionFilterMBS	808
* 8.4.3 Enabled as Boolean	808
* 8.4.5 FilterConnection(request as NSURLRequestMBS) as NSURLRequestMBS	808
– 8.5.1 class NSURLConnectionMBS	810
* 8.5.3 cancel	811
* 8.5.4 canHandleRequest(request as NSURLRequestMBS) as boolean	811
* 8.5.5 Constructor(request as NSURLRequestMBS)	811
* 8.5.6 Constructor(request as NSURLRequestMBS, startImmediately as boolean)	812
* 8.5.7 data as MemoryBlock	812
* 8.5.8 sendSynchronousRequest(request as NSURLRequestMBS, byref response as NSURLResponseMBS, byref error as NSErrorMBS) as Memoryblock	812

* 8.5.9 start	813
* 8.5.11 Handle as Integer	814
* 8.5.13 canAuthenticateAgainstProtectionSpace(ProtectionSpace as NSURLProtectionSpaceMBS) as boolean	814
* 8.5.14 didCancelAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)	814
* 8.5.15 didFailWithError(error as NSErrorMBS)	815
* 8.5.16 didFinishLoading	815
* 8.5.17 didReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)	815
* 8.5.18 didReceiveData(newData as MemoryBlock)	816
* 8.5.19 didReceiveResponse(response as NSURLResponseMBS)	816
* 8.5.20 didSendBodyData(bytesWritten as Int64, totalBytesWritten as Int64, totalBytesExpectedToWrite as Int64)	816
* 8.5.21 shouldUseCredentialStorage as boolean	817
* 8.5.22 willCacheResponse(cachedResponse as NSCachedURLResponseMBS) as NSCachedURLResponseMBS	817
* 8.5.23 willSendRequest(request as NSURLRequestMBS, redirectResponse as NSURLResponseMBS) as NSURLRequestMBS	817
* 8.5.24 willSendRequestForAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)	818
– 8.6.1 class NSURLCredentialMBS	820
* 8.6.3 Constructor	820
* 8.6.4 copy as NSURLCredentialMBS	820
* 8.6.5 credential(User as string, password as string, persistence as Integer = 0) as NSURLCredentialMBS	820
* 8.6.6 credentialWithPEM(Data as MemoryBlock, Password as String = "") as NSURLCredentialMBS	821
* 8.6.7 credentialWithPKCS12(Data as MemoryBlock, Password as String = "") as NSURLCredentialMBS	821
* 8.6.8 credentialWithTrustingServer(ProtectionSpace as NSURLProtectionSpaceMBS) as NSURLCredentialMBS	821
* 8.6.10 CertificateCount as Integer	822
* 8.6.11 Handle as Integer	822
* 8.6.12 HasIdentity as Boolean	822
* 8.6.13 hasPassword as Boolean	822
* 8.6.14 password as string	822
* 8.6.15 persistence as Integer	823
* 8.6.16 user as string	823
– 8.7.1 class NSURLCredentialStorageMBS	824
* 8.7.3 Constructor	824
* 8.7.4 defaultCredentialForProtectionSpace(space as NSURLProtectionSpaceMBS) as NSURLCredentialMBS	824

	79
* 8.7.5 sharedCredentialStorage as NSURLCredentialStorageMBS	824
* 8.7.7 Handle as Integer	824
– 8.8.1 class NSURLDownloadMBS	826
* 8.8.3 cancel	826
* 8.8.4 canResumeDownloadDecodedWithEncodingMIMEMType(MimeType as string) as boolean	826
* 8.8.5 Constructor(request as NSURLRequestMBS)	827
* 8.8.6 Constructor(resumeData as Memoryblock, path as folderitem)	827
* 8.8.7 Constructor(resumeData as Memoryblock, path as string)	827
* 8.8.8 request as NSURLRequestMBS	828
* 8.8.9 resumeData as Memoryblock	828
* 8.8.10 setDestination(path as folderitem, allowOverwrite as boolean)	828
* 8.8.11 setDestination(path as string, allowOverwrite as boolean)	829
* 8.8.13 Handle as Integer	829
* 8.8.14 deletesFileUponFailure as boolean	829
* 8.8.16 canAuthenticateAgainstProtectionSpace(protectionSpace as NSURLProtectionSpaceMBS) as boolean	830
* 8.8.17 decideDestinationWithSuggestedFilename(filename as string)	830
* 8.8.18 DidBegin	830
* 8.8.19 didCancelAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)	831
* 8.8.20 didCreateDestination(path as string, file as folderitem)	831
* 8.8.21 didFailWithError(error as NSErrorMBS)	831
* 8.8.22 DidFinish	832
* 8.8.23 didReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)	832
* 8.8.24 didReceiveDataOfLength(length as UInt64)	833
* 8.8.25 didReceiveResponse(response as NSURLResponseMBS)	833
* 8.8.26 shouldDecodeSourceDataOfMIMEMType(encodingType as string) as boolean	833
* 8.8.27 shouldUseCredentialStorage as boolean	834
* 8.8.28 willResumeWithResponse(response as NSURLResponseMBS, startingByte as Int64)	834
* 8.8.29 willSendRequest(request as NSURLRequestMBS, redirectResponse as NSURLResponseMBS) as NSURLRequestMBS	834
– 8.9.1 class NSURLMBS	836
* 8.9.3 checkResourceIsReachableAndReturnError as NSErrorMBS	836
* 8.9.4 Constructor(item as folderitem)	837
* 8.9.5 Constructor(scheme as string, host as string, path as string)	837
* 8.9.6 Constructor(url as string)	837
* 8.9.7 Constructor(url as string, baseURL as NSURLMBS)	838
* 8.9.8 copy as NSURLMBS	838

* 8.9.9 fileURLWithFileSystemRepresentation(path as string, isDirectory as boolean, relativeToURL as NSURLMBS) as NSURLMBS	838
* 8.9.10 fileURLWithPath(path as string) as NSURLMBS	838
* 8.9.11 fileURLWithPath(path as string, isDirectory as boolean) as NSURLMBS	839
* 8.9.12 fileURLWithPathComponents(components() as string) as NSURLMBS	839
* 8.9.13 getResourceValue(byref value as Variant, key as string, byref error as NSErrorMBS) as boolean	839
* 8.9.14 isEqual(other as NSURLMBS) as boolean	840
* 8.9.15 Items(byref error as NSErrorMBS, VisibleItemsOnly as boolean = false) as NSURLMBS()	840
* 8.9.16 Items(VisibleItemsOnly as boolean = false) as NSURLMBS()	840
* 8.9.17 mountedVolumeURLs(SkipHidden as boolean = true) as NSURLMBS()	840
* 8.9.18 NSThumbnail1024x1024SizeKey as string	841
* 8.9.19 NSURLAddedToDirectoryDateKey as string	841
* 8.9.20 NSURLAttributeModificationDateKey as string	841
* 8.9.21 NSURLContentAccessDateKey as string	841
* 8.9.22 NSURLContentModificationDateKey as string	842
* 8.9.23 NSURLCreationDateKey as string	842
* 8.9.24 NSURLCustomIconKey as string	842
* 8.9.25 NSURLDocumentIdentifierKey as string	842
* 8.9.26 NSURLEffectiveIconKey as string	843
* 8.9.27 NSURLFileAllocatedSizeKey as string	843
* 8.9.28 NSURLFileResourceIdentifierKey as string	843
* 8.9.29 NSURLFileResourceTypeBlockSpecial as string	844
* 8.9.30 NSURLFileResourceTypeCharacterSpecial as string	844
* 8.9.31 NSURLFileResourceTypeDirectory as string	844
* 8.9.32 NSURLFileResourceTypeKey as string	844
* 8.9.33 NSURLFileResourceTypeNamedPipe as string	844
* 8.9.34 NSURLFileResourceTypeRegular as string	845
* 8.9.35 NSURLFileResourceTypeSocket as string	845
* 8.9.36 NSURLFileResourceTypeSymbolicLink as string	845
* 8.9.37 NSURLFileResourceTypeUnknown as string	845
* 8.9.38 NSURLFileScheme as string	845
* 8.9.39 NSURLFileSecurityKey as string	846
* 8.9.40 NSURLFileSizeKey as string	846
* 8.9.41 NSURLGenerationIdentifierKey as string	846
* 8.9.42 NSURLHasHiddenExtensionKey as string	846
* 8.9.43 NSURLIsAliasFileKey as string	847
* 8.9.44 NSURLIsDirectoryKey as string	847
* 8.9.45 NSURLIsExcludedFromBackupKey as string	847
* 8.9.46 NSURLIsExecutableKey as string	847
* 8.9.47 NSURLIsHiddenKey as string	848

* 8.9.48 NSURLIsMountTriggerKey as string	848
* 8.9.49 NSURLIsPackageKey as string	848
* 8.9.50 NSURLIsReadableKey as string	848
* 8.9.51 NSURLIsRegularFileKey as string	849
* 8.9.52 NSURLIsSymbolicLinkKey as string	849
* 8.9.53 NSURLIsSystemImmutableKey as string	849
* 8.9.54 NSURLIsUbiquitousItemKey as string	849
* 8.9.55 NSURLIsUserImmutableKey as string	850
* 8.9.56 NSURLIsVolumeKey as string	850
* 8.9.57 NSURLIsWritableKey as string	850
* 8.9.58 NSURLKeysOfUnsetValueKey as string	851
* 8.9.59 NSURLLabelColorKey as string	851
* 8.9.60 NSURLLabelNumberKey as string	851
* 8.9.61 NSURLLinkCountKey as string	851
* 8.9.62 NSURLLocalizedLabelKey as string	852
* 8.9.63 NSURLLocalizedNameKey as string	852
* 8.9.64 NSURLLocalizedTypeDescriptionKey as string	852
* 8.9.65 NSURLNameKey as string	852
* 8.9.66 NSURLParentDirectoryURLKey as string	852
* 8.9.67 NSURLPathKey as string	853
* 8.9.68 NSURLPreferredIOBlockSizeKey as string	853
* 8.9.69 NSURLQuarantinePropertiesKey as string	853
* 8.9.70 NSURLTagNameKey as string	854
* 8.9.71 NSURLThumbnailDictionaryKey as string	854
* 8.9.72 NSURLThumbnailKey as string	854
* 8.9.73 NSURLTotalFileAllocatedSizeKey as string	854
* 8.9.74 NSURLTotalFileSizeKey as string	855
* 8.9.75 NSURLTypeIDentifierKey as string	855
* 8.9.76 NSURLUbiquitousItemContainerDisplayNameKey as string	855
* 8.9.77 NSURLUbiquitousItemDownloadingErrorKey as string	855
* 8.9.78 NSURLUbiquitousItemDownloadingStatusCurrent as string	856
* 8.9.79 NSURLUbiquitousItemDownloadingStatusDownloaded as string	856
* 8.9.80 NSURLUbiquitousItemDownloadingStatusKey as string	856
* 8.9.81 NSURLUbiquitousItemDownloadingStatusNotDownloaded as string	856
* 8.9.82 NSURLUbiquitousItemDownloadRequestedKey as string	857
* 8.9.83 NSURLUbiquitousItemHasUnresolvedConflictsKey as string	857
* 8.9.84 NSURLUbiquitousItemIsDownloadedKey as string	857
* 8.9.85 NSURLUbiquitousItemIsDownloadingKey as string	857
* 8.9.86 NSURLUbiquitousItemIsExcludedFromSyncKey as string	858
* 8.9.87 NSURLUbiquitousItemIsSharedKey as string	858
* 8.9.88 NSURLUbiquitousItemIsUploadedKey as string	858
* 8.9.89 NSURLUbiquitousItemIsUploadingKey as string	858

* 8.9.90 NSURLUbiquitousItemPercentDownloadedKey as string	859
* 8.9.91 NSURLUbiquitousItemPercentUploadedKey as string	859
* 8.9.92 NSURLUbiquitousItemUploadingErrorKey as string	859
* 8.9.93 NSURLUbiquitousSharedItemCurrentUserPermissionsKey as string	860
* 8.9.94 NSURLUbiquitousSharedItemCurrentUserRoleKey as string	860
* 8.9.95 NSURLUbiquitousSharedItemMostRecentEditorNameComponentsKey as string	860
* 8.9.96 NSURLUbiquitousSharedItemOwnerNameComponentsKey as string	860
* 8.9.97 NSURLUbiquitousSharedItemPermissionsReadOnly as string	860
* 8.9.98 NSURLUbiquitousSharedItemPermissionsReadWrite as string	861
* 8.9.99 NSURLUbiquitousSharedItemRoleOwner as string	861
* 8.9.100 NSURLUbiquitousSharedItemRoleParticipant as string	861
* 8.9.101 NSURLVolumeAvailableCapacityForImportantUsageKey as string	861
* 8.9.102 NSURLVolumeAvailableCapacityForOpportunisticUsageKey as string	862
* 8.9.103 NSURLVolumeAvailableCapacityKey as string	863
* 8.9.104 NSURLVolumeCreationDateKey as string	863
* 8.9.105 NSURLVolumeIdentifierKey as string	863
* 8.9.106 NSURLVolumeIsAutomountedKey as string	863
* 8.9.107 NSURLVolumeIsBrowsableKey as string	864
* 8.9.108 NSURLVolumeIsEjectableKey as string	864
* 8.9.109 NSURLVolumeIsEncryptedKey as string	864
* 8.9.110 NSURLVolumeIsInternalKey as string	864
* 8.9.111 NSURLVolumeIsJournalingKey as string	865
* 8.9.112 NSURLVolumeIsLocalKey as string	865
* 8.9.113 NSURLVolumeIsReadOnlyKey as string	866
* 8.9.114 NSURLVolumeIsRemovableKey as string	866
* 8.9.115 NSURLVolumeIsRootFileSystemKey as string	866
* 8.9.116 NSURLVolumeLocalizedFormatDescriptionKey as string	867
* 8.9.117 NSURLVolumeLocalizedNameKey as string	867
* 8.9.118 NSURLVolumeMaximumFileSizeKey as string	867
* 8.9.119 NSURLVolumeNameKey as string	867
* 8.9.120 NSURLVolumeResourceCountKey as string	867
* 8.9.121 NSURLVolumeSupportsAccessPermissionsKey as string	868
* 8.9.122 NSURLVolumeSupportsAdvisoryFileLockingKey as string	868
* 8.9.123 NSURLVolumeSupportsCasePreservedNamesKey as string	868
* 8.9.124 NSURLVolumeSupportsCaseSensitiveNamesKey as string	869
* 8.9.125 NSURLVolumeSupportsCompressionKey as string	869
* 8.9.126 NSURLVolumeSupportsExclusiveRenamingKey as string	869
* 8.9.127 NSURLVolumeSupportsExtendedSecurityKey as string	870
* 8.9.128 NSURLVolumeSupportsFileCloningKey as string	870
* 8.9.129 NSURLVolumeSupportsFileProtectionKey as string	870
* 8.9.130 NSURLVolumeSupportsHardLinksKey as string	870
* 8.9.131 NSURLVolumeSupportsImmutableFilesKey as string	870

* 8.9.132 NSURLVolumeSupportsJournalingKey as string	871
* 8.9.133 NSURLVolumeSupportsPersistentIDsKey as string	871
* 8.9.134 NSURLVolumeSupportsRenamingKey as string	871
* 8.9.135 NSURLVolumeSupportsRootDirectoryDatesKey as string	871
* 8.9.136 NSURLVolumeSupportsSparseFilesKey as string	871
* 8.9.137 NSURLVolumeSupportsSwapRenamingKey as string	872
* 8.9.138 NSURLVolumeSupportsSymbolicLinksKey as string	872
* 8.9.139 NSURLVolumeSupportsVolumeSizesKey as string	872
* 8.9.140 NSURLVolumeSupportsZeroRunsKey as string	872
* 8.9.141 NSURLVolumeTotalCapacityKey as string	873
* 8.9.142 NSURLVolumeURLForRemountingKey as string	873
* 8.9.143 NSURLVolumeURLKey as string	873
* 8.9.144 NSURLVolumeUUIDStringKey as string	873
* 8.9.145 pathComponents as string()	874
* 8.9.146 removeAllCachedResourceValues	874
* 8.9.147 removeCachedResourceValueForKey(key as string)	874
* 8.9.148 resourceValuesForKeys(keys() as string, byref error as NSErrorMBS) as Dictionary	874
* 8.9.149 resourceValuesForKeys(keys() as string, targetDelegate as ResourceValuesForKeysDelegateMBS, tag as Variant = nil, PrecacheIcons as boolean = false)	875
* 8.9.150 setResourceValue(value as Variant, key as string, byref error as NSErrorMBS) as boolean	876
* 8.9.151 setResourceValues(keyedValues as Dictionary, byref error as NSErrorMBS) as boolean	876
* 8.9.152 setTemporaryResourceValue(value as Variant, key as string)	877
* 8.9.153 startAccessingSecurityScopedResource as boolean	877
* 8.9.154 stopAccessingSecurityScopedResource	878
* 8.9.155 TagNames as string()	878
* 8.9.156 URLByAppendingPathComponent(pathComponent as string) as NSURLMBS	878
* 8.9.157 URLByAppendingPathComponent(pathComponent as string, isDirectory as boolean) as NSURLMBS	878
* 8.9.158 URLByAppendingPathExtension(PathExtension as string) as NSURLMBS	879
* 8.9.159 URLByDeletingLastPathComponent as NSURLMBS	879
* 8.9.160 URLByDeletingPathExtension as NSURLMBS	879
* 8.9.161 URLByResolvingSymlinksInPath as NSURLMBS	880
* 8.9.162 URLByStandardizingPath as NSURLMBS	880
* 8.9.163 URLsResourceValuesForKeys(URLs() as NSURLMBS, keys() as string, targetDelegate as URLsResourceValuesForKeysDelegateMBS, tag as Variant = nil, PrecacheIcons as boolean = false)	881
* 8.9.164 URLWithHandle(Handle as Integer) as NSURLMBS	881
* 8.9.165 URLWithItem(Item as FolderItem) as NSURLMBS	881
* 8.9.166 URLWithString(URL as string) as NSURLMBS	882
* 8.9.167 URLWithString(URL as string, baseURL as NSURLMBS) as NSURLMBS	882

* 8.9.169 absoluteString as String	882
* 8.9.170 absoluteURL as NSURLMBS	883
* 8.9.171 AddedToDirectoryDate as Date	883
* 8.9.172 AddedToDirectoryDateTime as DateTime	883
* 8.9.173 AttributeModificationDate as Date	883
* 8.9.174 AttributeModificationDateTime as DateTime	884
* 8.9.175 baseURL as NSURLMBS	884
* 8.9.176 ContentAccessDate as Date	884
* 8.9.177 ContentAccessDateTime as DateTime	884
* 8.9.178 ContentModificationDate as Date	885
* 8.9.179 ContentModificationDateTime as DateTime	885
* 8.9.180 CreationDate as Date	885
* 8.9.181 CreationDateTime as DateTime	885
* 8.9.182 DocumentIdentifier as String	886
* 8.9.183 EffectiveIcon as Variant	886
* 8.9.184 filePathURL as NSURLMBS	886
* 8.9.185 fileReferenceURL as NSURLMBS	886
* 8.9.186 FileResourceIdentifier as String	887
* 8.9.187 FileResourceType as String	887
* 8.9.188 fileSystemRepresentation as String	888
* 8.9.189 fragment as String	888
* 8.9.190 GenerationIdentifier as String	888
* 8.9.191 Handle as Integer	889
* 8.9.192 HasHiddenExtension as Boolean	889
* 8.9.193 host as String	889
* 8.9.194 IsAlias as Boolean	889
* 8.9.195 IsDirectory as Boolean	890
* 8.9.196 IsExcludedFromBackup as Boolean	890
* 8.9.197 IsExecutable as Boolean	890
* 8.9.198 isFileReferenceURL as Boolean	890
* 8.9.199 isFileURL as Boolean	891
* 8.9.200 IsHidden as Boolean	891
* 8.9.201 IsMountTrigger as Boolean	891
* 8.9.202 IsPackage as Boolean	891
* 8.9.203 IsReadable as Boolean	891
* 8.9.204 IsRegularFile as Boolean	893
* 8.9.205 IsSymbolicLink as Boolean	893
* 8.9.206 IsSystemImmutable as Boolean	893
* 8.9.207 IsUbiquitousItem as Boolean	893
* 8.9.208 IsUserImmutable as Boolean	893
* 8.9.209 IsVolume as Boolean	894
* 8.9.210 IsWritable as Boolean	894

* 8.9.211 Item as FolderItem	894
* 8.9.212 LabelColor as Variant	894
* 8.9.213 LabelNumber as Integer	894
* 8.9.214 lastPathComponent as String	895
* 8.9.215 LinkCount as Integer	895
* 8.9.216 LocalizedLabel as String	895
* 8.9.217 LocalizedName as String	895
* 8.9.218 LocalizedTypeDescription as String	896
* 8.9.219 Name as String	896
* 8.9.220 parameterString as String	896
* 8.9.221 ParentDirectoryURL as NSURLMBS	896
* 8.9.222 password as String	896
* 8.9.223 path as String	897
* 8.9.224 pathExtension as String	897
* 8.9.225 port as Integer	898
* 8.9.226 PreferredIOBlockSize as Integer	898
* 8.9.227 QuarantineProperties as Dictionary	898
* 8.9.228 query as String	899
* 8.9.229 relativePath as String	899
* 8.9.230 relativeString as String	899
* 8.9.231 resourceSpecifier as String	899
* 8.9.232 scheme as String	900
* 8.9.233 standardizedURL as NSURLMBS	900
* 8.9.234 TypeIdentifier as String	900
* 8.9.235 UbiquitousItemContainerDisplayName as String	900
* 8.9.236 UbiquitousItemDownloadingError as NSErrorMBS	901
* 8.9.237 UbiquitousItemDownloadingStatus as String	901
* 8.9.238 UbiquitousItemDownloadRequested as Boolean	901
* 8.9.239 UbiquitousItemHasUnresolvedConflicts as Boolean	901
* 8.9.240 UbiquitousItemIsDownloaded as Boolean	901
* 8.9.241 UbiquitousItemIsDownloading as Boolean	902
* 8.9.242 UbiquitousItemIsExcludedFromSync as Boolean	902
* 8.9.243 UbiquitousItemIsShared as Boolean	902
* 8.9.244 UbiquitousItemIsUploaded as Boolean	902
* 8.9.245 UbiquitousItemIsUploading as Boolean	902
* 8.9.246 UbiquitousItemUploadingError as NSErrorMBS	902
* 8.9.247 user as String	903
* 8.9.248 VolumeIdentifier as String	903
* 8.9.249 VolumeURL as NSURLMBS	903
* 8.9.251 ResourceValuesForKeysDelegateMBS(URL as NSURLMBS, keys() as String, Values as Dictionary, Error as NSErrorMBS, tag as variant)	904

* 8.9.252 URLsResourceValuesForKeysDelegateMBS(URLs() as NSURLMBS, keys() as String, Values() as Dictionary, Errors() as NSErrorMBS, tag as variant)	904
– 8.10.1 class NSURLProtectionSpaceMBS	905
* 8.10.3 authenticationMethod as string	905
* 8.10.4 Constructor	905
* 8.10.5 host as string	905
* 8.10.6 isProxy as boolean	905
* 8.10.7 NSURLAuthenticationMethodClientCertificate as String	906
* 8.10.8 NSURLAuthenticationMethodDefault as String	906
* 8.10.9 NSURLAuthenticationMethodHTMLForm as String	906
* 8.10.10 NSURLAuthenticationMethodHTTPBasic as String	906
* 8.10.11 NSURLAuthenticationMethodHTTPDigest as String	906
* 8.10.12 NSURLAuthenticationMethodNegotiate as String	906
* 8.10.13 NSURLAuthenticationMethodNTLM as String	907
* 8.10.14 NSURLAuthenticationMethodServerTrust as String	907
* 8.10.15 NSURLProtectionSpaceFTP as String	907
* 8.10.16 NSURLProtectionSpaceFTPProxy as String	907
* 8.10.17 NSURLProtectionSpaceHTTP as String	907
* 8.10.18 NSURLProtectionSpaceHTTPProxy as String	907
* 8.10.19 NSURLProtectionSpaceHTTPS as String	908
* 8.10.20 NSURLProtectionSpaceHTTPSProxy as String	908
* 8.10.21 NSURLProtectionSpaceSOCKSProxy as String	908
* 8.10.22 port as Integer	908
* 8.10.23 protocol as string	908
* 8.10.24 proxyType as string	908
* 8.10.25 realm as string	909
* 8.10.26 receivesCredentialSecurely as boolean	909
* 8.10.28 Handle as Integer	909
– 8.11.1 class NSURLRequestMBS	910
* 8.11.3 allHTTPHeaderFields as Dictionary	910
* 8.11.4 Constructor(url as string)	911
* 8.11.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double)	911
* 8.11.6 copy as NSURLRequestMBS	911
* 8.11.7 mutableCopy as NSMutableURLRequestMBS	912
* 8.11.8 requestWithHandle(Handle as Integer) as NSURLRequestMBS	912
* 8.11.9 requestWithURL(url as string) as NSURLRequestMBS	912
* 8.11.10 requestWithURL(url as string, cachePolicy as Integer, timeoutInterval as Double) as NSURLRequestMBS	913
* 8.11.11 valueForHTTPHeaderField(field as string) as string	913
* 8.11.13 attribution as Integer	914
* 8.11.14 cachePolicy as Integer	914

	87
* 8.11.15 Handle as Integer	914
* 8.11.16 HTTPBody as memoryblock	915
* 8.11.17 HTTPMethod as string	915
* 8.11.18 HTTPShouldHandleCookies as boolean	915
* 8.11.19 HTTPShouldUsePipelining as boolean	915
* 8.11.20 isHTTPRequest as boolean	916
* 8.11.21 mainDocumentURL as string	916
* 8.11.22 networkServiceType as Integer	916
* 8.11.23 timeoutInterval as Double	916
* 8.11.24 URL as string	917
– 8.12.1 class NSURLResponseMBS	920
* 8.12.3 allHeaderFields as Dictionary	920
* 8.12.4 Constructor(URL as string, MIMEType as string, expectedContentLength as Integer, textEncodingName as string)	920
* 8.12.5 copy as NSURLResponseMBS	921
* 8.12.6 expectedContentLength as int64	921
* 8.12.7 isHTTPResponse as boolean	921
* 8.12.8 localizedStringForStatusCode(statusCode as Integer) as string	921
* 8.12.9 MIMEType as string	921
* 8.12.10 statusCode as Integer	922
* 8.12.11 suggestedFilename as string	922
* 8.12.12 textEncodingName as string	922
* 8.12.13 URL as string	923
* 8.12.15 Handle as Integer	923

• 4 Cocoa	147
– 4.30.1 class NSUserDefaultsMBS	400
* 4.30.3 addSuiteNamed(suiteName as string)	401
* 4.30.4 arrayForKey(key as string) as Variant()	401
* 4.30.5 Constructor	402
* 4.30.6 Constructor(username as string)	402
* 4.30.7 dictionaryRepresentation as dictionary	403
* 4.30.8 NSArgumentDomain as string	403
* 4.30.9 NSGlobalDomain as string	403
* 4.30.10 NSRegistrationDomain as string	404
* 4.30.11 NSUserDefaultsDidChangeNotification as string	404
* 4.30.12 objectIsForcedForKey(key as string) as boolean	404
* 4.30.13 objectIsForcedForKey(key as string, domain as string) as boolean	404
* 4.30.14 persistentDomainForName(domainName as string) as dictionary	405
* 4.30.15 persistentDomainNames as string()	405
* 4.30.16 registerDefaults(dic as dictionary)	405
* 4.30.17 removeObjectForKey(defaultName as string)	406
* 4.30.18 removePersistentDomainForName(domainName as string)	406
* 4.30.19 removeSuiteNamed(suiteName as string)	407
* 4.30.20 removeVolatileDomainForName(domainName as string)	407
* 4.30.21 resetStandardUserDefaults	407
* 4.30.22 setArrayValue(key as string, values() as Variant)	407
* 4.30.23 setBoolValue(key as string, value as boolean)	408
* 4.30.24 setDataValue(key as string, value as memoryblock)	408
* 4.30.25 setDictionaryValue(key as string, value as dictionary)	408
* 4.30.26 setDoubleValue(key as string, value as Double)	409
* 4.30.27 setFileValue(key as string, value as folderitem)	409
* 4.30.28 setFloatValue(key as string, value as single)	410
* 4.30.29 setIntegerValue(key as string, value as Integer)	410
* 4.30.30 setPersistentDomain(domain as dictionary, domainName as string)	410
* 4.30.31 setStringArrayValue(key as string, values() as string)	411
* 4.30.32 setStringValue(key as string, value as string)	411
* 4.30.33 setURLValue(key as string, value as string)	411
* 4.30.34 setVariantValue(key as string, value as Variant)	412
* 4.30.35 setVolatileDomain(domain as dictionary, domainName as string)	412
* 4.30.36 standardUserDefaults as NSUserDefaultsMBS	412
* 4.30.37 stringArrayForKey(key as string) as string()	413
* 4.30.38 synchronize as boolean	413
* 4.30.39 volatileDomainForName(domainName as string) as dictionary	414
* 4.30.40 volatileDomainNames as string()	414
* 4.30.42 boolForKey(key as string) as boolean	414

* 4.30.43	dataForKey(key as string) as memoryblock	415
* 4.30.44	dictionaryForKey(key as string) as dictionary	415
* 4.30.45	doubleForKey(key as string) as Double	416
* 4.30.46	fileForKey(key as string) as folderitem	416
* 4.30.47	floatForKey(key as string) as single	417
* 4.30.48	integerForKey(key as string) as Integer	417
* 4.30.49	stringForKey(key as string) as string	417
* 4.30.50	URLForKey(key as string) as string	418
* 4.30.51	variantForKey(key as string) as Variant	418
– 4.31.1	class NSUUIDMBS	419
* 4.31.3	Available as boolean	419
* 4.31.4	Constructor	419
* 4.31.5	Constructor(UUID as MemoryBlock)	419
* 4.31.6	Constructor(UUID as String)	420
* 4.31.7	copy as NSUUIDMBS	420
* 4.31.8	isEqual(other as NSUUIDMBS) as boolean	420
* 4.31.9	Operator__Compare(other as NSUUIDMBS) as Integer	421
* 4.31.10	UUID as NSUUIDMBS	421
* 4.31.12	data as MemoryBlock	422
* 4.31.13	Handle as Integer	422
* 4.31.14	UUIDString as String	422
– 4.32.1	class NSViewControllerMBS	423
* 4.32.3	available as boolean	423
* 4.32.4	Constructor	423
* 4.32.5	contentViewController(window as NSWindowMBS) as NSViewControllerMBS	424
* 4.32.6	dismissViewController(ViewController as NSViewControllerMBS)	424
* 4.32.7	loadView	424
* 4.32.8	presentViewControllerAsModalWindow(ViewController as NSViewControllerMBS)	425
* 4.32.9	presentViewControllerAsPopover(ViewController as NSViewControllerMBS, RelativeToRect as NSRectMBS, positioningView as NSViewMBS, preferredEdge as Integer, behavior as Integer)	425
* 4.32.10	presentViewControllerAsSheet(ViewController as NSViewControllerMBS)	425
* 4.32.11	windowWithContentViewController(ViewController as NSViewControllerMBS) as NSWindowMBS	426
* 4.32.13	className as string	426
* 4.32.14	classPath as string	426
* 4.32.15	Identifier as String	426
* 4.32.16	representedObject as Variant	427
* 4.32.17	Title as string	427
* 4.32.18	view as NSViewMBS	427
* 4.32.19	viewLoaded as Boolean	427

– 4.33.1 class <code>NSViewMBS</code>	429
* 4.33.3 <code>addSubview(subview as NSViewMBS)</code>	429
* 4.33.4 <code>addSubview(subview as NSViewMBS, positioned as Integer, relativeToView as NSViewMBS)</code>	430
* 4.33.5 <code>addToolTipRect(rect as NSRectMBS, tooltip as NSViewToolTipMBS)</code>	430
* 4.33.6 <code>ancestorSharedWithView(view as NSViewMBS) as NSViewMBS</code>	431
* 4.33.7 <code>animator as NSViewMBS</code>	431
* 4.33.8 <code>backgroundFilters as variant()</code>	431
* 4.33.9 <code>beginDraggingSessionWithItems(items() as Variant, e as NSEventMBS, source as NSViewMBS) as Variant</code>	432
* 4.33.10 <code>Constructor</code>	432
* 4.33.11 <code>Constructor(Handle as Integer)</code>	432
* 4.33.12 <code>Constructor(left as Double, top as Double, width as Double, height as Double)</code>	433
* 4.33.13 <code>contentFilters as variant()</code>	433
* 4.33.14 <code>convertPointFromView(point as NSPointMBS, View as NSViewMBS) as NSPointMBS</code>	434
* 4.33.15 <code>convertPointToView(point as NSPointMBS, View as NSViewMBS) as NSPointMBS</code>	434
* 4.33.16 <code>convertRectFromView(rect as NSRectMBS, View as NSViewMBS) as NSRectMBS</code>	434
* 4.33.17 <code>convertRectToView(rect as NSRectMBS, View as NSViewMBS) as NSRectMBS</code>	434
* 4.33.18 <code>convertSizeFromView(Size as NSSizeMBS, View as NSViewMBS) as NSSizeMBS</code>	435
* 4.33.19 <code>convertSizeToView(Size as NSSizeMBS, View as NSViewMBS) as NSSizeMBS</code>	435
* 4.33.20 <code>dataWithEPSInsideRect(left as Double, top as Double, width as Double, height as Double) as Memoryblock</code>	435
* 4.33.21 <code>dataWithPDFInsideRect(left as Double, top as Double, width as Double, height as Double) as Memoryblock</code>	436
* 4.33.22 <code>dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock</code>	436
* 4.33.23 <code>dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NSSizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)</code>	437
* 4.33.24 <code>drawFocusRingMask</code>	437
* 4.33.25 <code>enclosingMenuItem as Variant</code>	438
* 4.33.26 <code>enclosingScrollView as Variant</code>	438
* 4.33.27 <code>focusRingMaskBounds as NSRectMBS</code>	438
* 4.33.28 <code>isDescendantOf(view as NSViewMBS) as boolean</code>	438
* 4.33.29 <code>makeBackingLayer as Variant</code>	439
* 4.33.30 <code>nextValidKeyView as NSViewMBS</code>	439
* 4.33.31 <code>noteFocusRingMaskChanged</code>	439
* 4.33.32 <code>NSViewBoundsDidChangeNotification as string</code>	440
* 4.33.33 <code>NSViewDidUpdateTrackingAreasNotification as string</code>	440
* 4.33.34 <code>NSViewFocusDidChangeNotification as string</code>	440
* 4.33.35 <code>NSViewFrameDidChangeNotification as string</code>	441

* 4.33.36	NSViewGlobalFrameDidChangeNotification as string	441
* 4.33.37	pageFooter as NSAttributedStringMBS	441
* 4.33.38	pageHeader as NSAttributedStringMBS	442
* 4.33.39	previousKeyView as NSViewMBS	442
* 4.33.40	previousValidKeyView as NSViewMBS	442
* 4.33.41	print	442
* 4.33.42	registeredDraggedTypes as string()	443
* 4.33.43	registerForDraggedTypes(Types() as string)	443
* 4.33.44	removeAllToolTips	444
* 4.33.45	removeFromSuperview	444
* 4.33.46	removeFromSuperviewWithoutNeedingDisplay	444
* 4.33.47	RenderImage(subviews as boolean = false, flipped as boolean = false) as variant	444
* 4.33.48	replaceSubview(oldView as NSViewMBS, newView as NSViewMBS)	445
* 4.33.49	rotateByAngle(angle as Double)	445
* 4.33.50	scaleUnitSquareToSize(size as NSSizeMBS)	445
* 4.33.51	Screenshot as Picture	446
* 4.33.52	setBackgroundFilters(Filters() as variant)	446
* 4.33.53	setBoundsOrigin(origin as NSPointMBS)	446
* 4.33.54	setBoundsOrigin(x as Double, y as Double)	446
* 4.33.55	setBoundsSize(size as NSSizeMBS)	447
* 4.33.56	setBoundsSize(width as Double, height as Double)	447
* 4.33.57	setContentFilters(Filters() as variant)	447
* 4.33.58	setFocus	447
* 4.33.59	setFrameOrigin(origin as NSPointMBS)	447
* 4.33.60	setFrameOrigin(x as Double, y as Double)	448
* 4.33.61	setFrameSize(size as NSSizeMBS)	448
* 4.33.62	setFrameSize(width as Double, height as Double)	448
* 4.33.63	subviews(recursive as boolean = false) as NSViewMBS()	448
* 4.33.64	unregisterDraggedTypes	449
* 4.33.66	acceptsTouchEvents as boolean	449
* 4.33.67	allowsVibrancy as Boolean	449
* 4.33.68	alphaValue as Double	449
* 4.33.69	autoresizesSubviews as boolean	450
* 4.33.70	autoresizingMask as Integer	450
* 4.33.71	bounds as NSRectMBS	450
* 4.33.72	boundsRotation as Double	451
* 4.33.73	canBecomeKeyView as boolean	451
* 4.33.74	canDraw as boolean	451
* 4.33.75	canDrawConcurrently as boolean	451
* 4.33.76	className as string	452
* 4.33.77	classPath as string	452
* 4.33.78	clipsToBounds as Boolean	452

* 4.33.79 focusRingType as Integer	453
* 4.33.80 frame as NSRectMBS	453
* 4.33.81 frameCenterRotation as Double	453
* 4.33.82 frameHeight as Double	454
* 4.33.83 frameLeft as Double	454
* 4.33.84 frameRotation as Double	454
* 4.33.85 frameTop as Double	455
* 4.33.86 frameWidth as Double	455
* 4.33.87 identifier as string	455
* 4.33.88 isFlipped as Boolean	456
* 4.33.89 isHidden as Boolean	456
* 4.33.90 isHiddenOrHasHiddenAncestor as Boolean	456
* 4.33.91 isOpaque as Boolean	456
* 4.33.92 isRotatedFromBase as Boolean	456
* 4.33.93 isRotatedOrScaledFromBase as Boolean	457
* 4.33.94 layer as Variant	457
* 4.33.95 layerUsesCoreImageFilters as Boolean	457
* 4.33.96 needsDisplay as Boolean	458
* 4.33.97 nextKeyView as NSViewMBS	458
* 4.33.98 opaqueAncestor as NSViewMBS	458
* 4.33.99 RetainCount as Integer	458
* 4.33.100 superview as NSViewMBS	458
* 4.33.101 toolTip as string	459
* 4.33.102 userInteractionEnabled as Boolean	459
* 4.33.103 visibleRect as NSRectMBS	459
* 4.33.104 wantsDefaultClipping as boolean	460
* 4.33.105 wantsLayer as Boolean	460
* 4.33.106 wantsRestingTouches as boolean	460
* 4.33.107 window as NSWindowMBS	461
* 4.33.108 compositingFilter as variant	461
– 4.34.1 class NSViewTooltipMBS	463
* 4.34.3 Constructor	463
* 4.34.5 Text as String	463
* 4.34.7 stringForToolTip(point as NSPointMBS) as string	463
– 4.35.1 class NSWindowControllerMBS	465
* 4.35.3 close	465
* 4.35.4 Constructor(win as NSWindowMBS)	465
* 4.35.5 Constructor(windowNibName as string)	466
* 4.35.6 showWindow	466
* 4.35.7 synchronizeWindowTitleWithDocumentName	466
* 4.35.9 className as string	466

* 4.35.10 classPath as string	467
* 4.35.11 shouldCascadeWindows as boolean	467
* 4.35.12 shouldCloseDocument as boolean	467
* 4.35.13 window as NSWindowMBS	467
* 4.35.14 windowFrameAutosaveName as string	468
* 4.35.15 windowNibName as string	468
* 4.35.16 windowNibPath as string	468
– 4.36.1 class NSWindowMBS	469
* 4.36.3 addChildWindow(win as DesktopWindow, order as integer)	470
* 4.36.4 addChildWindow(win as NSWindowMBS, order as Integer)	470
* 4.36.5 addChildWindow(win as window, order as Integer)	471
* 4.36.6 addTabbedWindow(win as NSWindowMBS, ordered as Integer)	471
* 4.36.7 animator as NSWindowMBS	472
* 4.36.8 areCursorRectsEnabled as boolean	472
* 4.36.9 attachedSheet as NSWindowMBS	472
* 4.36.10 autorecalculatesContentBorderThicknessForEdge(edge as Integer) as boolean	473
* 4.36.11 becomeKeyWindow	473
* 4.36.12 becomeMainWindow	473
* 4.36.13 cacheImageInRect(r as NSRectMBS)	473
* 4.36.14 Center	474
* 4.36.15 childWindows as NSWindowMBS()	474
* 4.36.16 ClearFocus	474
* 4.36.17 Close	474
* 4.36.18 Constructor(w as DesktopWindow)	475
* 4.36.19 Constructor(w as window)	475
* 4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false)	475
* 4.36.21 contentBorderThicknessForEdge(edge as Integer) as Double	476
* 4.36.22 contentRectForFrameRect(windowFrame as NSRectMBS) as NSRectMBS	477
* 4.36.23 contentRectForFrameRect(windowFrame as NSRectMBS, styleMask as UInt32) as NSRectMBS	477
* 4.36.24 convertBaseToScreen(p as NSPointMBS) as NSPointMBS	478
* 4.36.25 convertScreenToBase(p as NSPointMBS) as NSPointMBS	478
* 4.36.26 dataWithEPSInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.27 dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.28 deminiaturize	478
* 4.36.29 disableCursorRects	479
* 4.36.30 disableFlushWindow	479
* 4.36.31 disableScreenUpdatesUntilFlush	479
* 4.36.32 disableSnapshotRestoration	479
* 4.36.33 discardCachedImage	480

* 4.36.34	discardCursorRects	480
* 4.36.35	display	480
* 4.36.36	displayIfNeeded	480
* 4.36.37	dockTile as Variant	481
* 4.36.38	dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NS-SizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)	481
* 4.36.39	enableCursorRects	481
* 4.36.40	enableFlushWindow	481
* 4.36.41	enableSnapshotRestoration	482
* 4.36.42	endEditingFor(anObject as object = nil)	482
* 4.36.43	fieldEditor(createFlag as boolean = True, forObject as object = nil) as Variant	482
* 4.36.44	firstResponder as NSResponderMBS	483
* 4.36.45	flushWindow	484
* 4.36.46	flushWindowIfNeeded	484
* 4.36.47	frameRectForContentRect(windowContent as NSRectMBS) as NSRectMBS	484
* 4.36.48	frameRectForContentRect(windowContentRect as NSRectMBS, styleMask as UInt32) as NSRectMBS	485
* 4.36.49	GetFrame(byref left as Double, byref top as Double, byref width as Double, byref height as Double)	485
* 4.36.50	gState as Integer	485
* 4.36.51	Hide	485
* 4.36.52	inLiveResize as boolean	486
* 4.36.53	invalidateCursorRectsForView(View as NSViewMBS)	486
* 4.36.54	invalidateRestorableState	486
* 4.36.55	invalidateShadow	486
* 4.36.56	keyDown(e as NSEventMBS)	486
* 4.36.57	makeFirstResponder(r as NSResponderMBS) as boolean	486
* 4.36.58	makeKeyAndOrderFront	487
* 4.36.59	makeKeyWindow	488
* 4.36.60	makeMainWindow	488
* 4.36.61	mergeAllWindows	488
* 4.36.62	minFrameWidthWithTitle(WindowTitle as string, styleMask as UInt32) as Double	488
* 4.36.63	miniaturize	489
* 4.36.64	moveTabToNewWindow	489
* 4.36.65	NSDockWindowLevel as Integer	489
* 4.36.66	NSFloatingWindowLevel as Integer	489
* 4.36.67	NSMainMenuWindowLevel as Integer	489
* 4.36.68	NSModalPanelWindowLevel as Integer	490
* 4.36.69	NSNormalWindowLevel as Integer	490
* 4.36.70	NSPopUpMenuWindowLevel as Integer	490

* 4.36.71 NSScreenSaverWindowLevel as Integer	490
* 4.36.72 NSStatusWindowLevel as Integer	490
* 4.36.73 NSSubmenuWindowLevel as Integer	491
* 4.36.74 NSTornOffMenuWindowLevel as Integer	491
* 4.36.75 NSWindowDidBecomeKeyNotification as string	491
* 4.36.76 NSWindowDidBecomeMainNotification as string	491
* 4.36.77 NSWindowDidChangeScreenNotification as string	491
* 4.36.78 NSWindowDidChangeScreenProfileNotification as string	492
* 4.36.79 NSWindowDidDeminiaturizeNotification as string	492
* 4.36.80 NSWindowDidEndLiveResizeNotification as string	492
* 4.36.81 NSWindowDidEndSheetNotification as string	493
* 4.36.82 NSWindowDidEnterFullScreenNotification as string	493
* 4.36.83 NSWindowDidEnterVersionBrowserNotification as string	493
* 4.36.84 NSWindowDidExitFullScreenNotification as string	494
* 4.36.85 NSWindowDidExitVersionBrowserNotification as string	494
* 4.36.86 NSWindowDidExposeNotification as string	494
* 4.36.87 NSWindowDidMiniaturizeNotification as string	494
* 4.36.88 NSWindowDidMoveNotification as string	495
* 4.36.89 NSWindowDidResignKeyNotification as string	495
* 4.36.90 NSWindowDidResignMainNotification as string	495
* 4.36.91 NSWindowDidResizeNotification as string	496
* 4.36.92 NSWindowDidUpdateNotification as string	496
* 4.36.93 NSWindowWillBeginSheetNotification as string	496
* 4.36.94 NSWindowWillCloseNotification as string	496
* 4.36.95 NSWindowWillEnterFullScreenNotification as string	497
* 4.36.96 NSWindowWillEnterVersionBrowserNotification as string	497
* 4.36.97 NSWindowWillExitFullScreenNotification as string	497
* 4.36.98 NSWindowWillExitVersionBrowserNotification as string	497
* 4.36.99 NSWindowWillMiniaturizeNotification as string	498
* 4.36.100 NSWindowWillMoveNotification as string	498
* 4.36.101 NSWindowWillStartLiveResizeNotification as string	498
* 4.36.102 orderBack	498
* 4.36.103 orderFront	499
* 4.36.104 orderFrontRegardless	499
* 4.36.105 orderOut	499
* 4.36.106 PerformClose	499
* 4.36.107 performMiniaturize	500
* 4.36.108 performWindowDragWithEvent(event as NSEventMBS)	500
* 4.36.109 performZoom	500
* 4.36.110 print	500
* 4.36.111 registerForDraggedTypes(Types() as string)	501
* 4.36.112 removeChildWindow(win as NSWindowMBS)	501

* 4.36.113 removeChildWindow(win as window)	501
* 4.36.114 removeFrameUsingName(name as string)	501
* 4.36.115 resetCursorRects	502
* 4.36.116 resignKeyWindow	502
* 4.36.117 resignMainWindow	502
* 4.36.118 resizeFlags as Integer	502
* 4.36.119 restoreCachedImage	503
* 4.36.121 saveFrameUsingName(s as String)	503
* 4.36.122 selectKeyViewFollowingView(view as NSViewMBS)	503
* 4.36.123 selectKeyViewPrecedingView(view as NSViewMBS)	504
* 4.36.124 selectNextKeyView	504
* 4.36.125 selectNextTab	504
* 4.36.126 selectPreviousKeyView	504
* 4.36.127 selectPreviousTab	505
* 4.36.128 sendEvent(e as NSEventMBS)	505
* 4.36.129 setAutorecalculatesContentBorderThickness(flag as boolean, edge as Integer)	505
* 4.36.130 setBottomCornerRounded(flag as boolean)	506
* 4.36.131 setContentBorderThickness(thickness as Double, edge as Integer)	506
* 4.36.132 setContentSize(size as NSSizeMBS)	507
* 4.36.133 setFrame(frameRect as NSRectMBS)	507
* 4.36.134 setFrame(frameRect as NSRectMBS, display as boolean)	507
* 4.36.135 setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)	508
* 4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double)	508
* 4.36.137 setFrameAutosaveName(name as String) as boolean	508
* 4.36.138 setFrameFromString(s as String)	509
* 4.36.139 setFrameOrigin(point as NSPointMBS)	509
* 4.36.140 setFrameTopLeftPoint(point as NSPointMBS)	509
* 4.36.141 setFrameUsingName(name as String, force as boolean = false) as boolean	510
* 4.36.142 setRestorationClass	510
* 4.36.143 setTitleWithRepresentedFile(filename as folderitem)	510
* 4.36.144 setTitleWithRepresentedFilename(filename as string)	510
* 4.36.145 Show	511
* 4.36.146 standardWindowButton(button as Integer) as Variant	511
* 4.36.147 standardWindowButton(button as Integer, StyleMask as Integer) as Variant	511
* 4.36.148 stringWithSavedFrame as String	512
* 4.36.149 tabbedWindows as NSWindowMBS()	512
* 4.36.150 toggleFullScreen	513
* 4.36.151 toggleTabBar	513
* 4.36.152 toggleToolBarShown	513
* 4.36.153 toolbarview as NSViewMBS	514
* 4.36.154 unregisterDraggedTypes	514
* 4.36.155 update	514

* 4.36.156 useOptimizedDrawing(value as boolean)	514
* 4.36.157 WindowHandle as Integer	515
* 4.36.158 windowNumberAtPoint(x as Double, y as Double, belowWindowWithWindowNumber as Integer = 0) as Integer	515
* 4.36.159 windowNumbersWithOptions(options as Integer = 0) as Integer()	515
* 4.36.160 zoom	516
* 4.36.162 acceptsMouseMovedEvents as boolean	516
* 4.36.163 allowsAutomaticWindowTabbing as Boolean	516
* 4.36.164 allowsConcurrentViewDrawing as boolean	517
* 4.36.165 allowsToolTipsWhenApplicationIsInactive as boolean	517
* 4.36.166 alphaValue as Double	517
* 4.36.167 animationBehavior as Integer	518
* 4.36.168 aspectRatio as NSSizeMBS	518
* 4.36.169 Autodisplay as boolean	518
* 4.36.170 backgroundColor as NSColorMBS	519
* 4.36.171 backingLocation as Integer	519
* 4.36.172 backingScaleFactor as Double	519
* 4.36.173 backingType as Integer	520
* 4.36.174 canBecomeKeyWindow as boolean	520
* 4.36.175 canBecomeMainWindow as boolean	520
* 4.36.176 canBecomeVisibleWithoutLogin as boolean	521
* 4.36.177 canHide as boolean	521
* 4.36.178 canStoreColor as boolean	521
* 4.36.179 className as string	521
* 4.36.180 classPath as string	522
* 4.36.181 collectionBehavior as Integer	522
* 4.36.182 colorSpace as NSColorSpaceMBS	522
* 4.36.183 contentAspectRatio as NSSizeMBS	522
* 4.36.184 contentLayoutRect as NSRectMBS	522
* 4.36.185 contentMaxSize as NSSizeMBS	523
* 4.36.186 contentMinSize as NSSizeMBS	523
* 4.36.187 contentResizeIncrements as NSSizeMBS	523
* 4.36.188 contentView as NSViewMBS	523
* 4.36.189 currentEvent as NSEventMBS	523
* 4.36.190 deepestScreen as NSScreenMBS	524
* 4.36.191 depthLimit as Integer	524
* 4.36.192 displaysWhenScreenProfileChanges as boolean	524
* 4.36.193 frame as NSRectMBS	524
* 4.36.194 hasDynamicDepthLimit as boolean	525
* 4.36.195 hasShadow as boolean	525
* 4.36.196 Height as Double	525
* 4.36.197 hidesOnDeactivate as boolean	525

* 4.36.198 identifier as string	525
* 4.36.199 ignoresMouseEvents as boolean	526
* 4.36.200 initialFirstResponder as NSViewMBS	526
* 4.36.201 isDocumentEdited as boolean	526
* 4.36.202 isExcludedFromWindowsMenu as boolean	527
* 4.36.203 isFlushWindowDisabled as boolean	527
* 4.36.204 isKeyWindow as boolean	528
* 4.36.205 isMainWindow as boolean	528
* 4.36.206 isMiniaturized as boolean	528
* 4.36.207 isMovableByWindowBackground as boolean	528
* 4.36.208 isOnActiveSpace as boolean	529
* 4.36.209 isOneShot as boolean	529
* 4.36.210 isOpaque as boolean	529
* 4.36.211 isSheet as boolean	529
* 4.36.212 isZoomed as boolean	529
* 4.36.213 Left as Double	530
* 4.36.214 Level as Integer	530
* 4.36.215 maxSize as NSSizeMBS	530
* 4.36.216 miniwindowImage as Variant	530
* 4.36.217 miniwindowTitle as String	531
* 4.36.218 minSize as NSSizeMBS	531
* 4.36.219 Movable as boolean	531
* 4.36.220 parentWindow as NSWindowMBS	531
* 4.36.221 preferredBackingLocation as Integer	532
* 4.36.222 preservesContentDuringLiveResize as boolean	532
* 4.36.223 preventsApplicationTerminationWhenModal as boolean	532
* 4.36.224 representedFile as folderitem	532
* 4.36.225 representedFilename as string	533
* 4.36.226 representedURL as string	533
* 4.36.227 resizeIncrements as NSSizeMBS	534
* 4.36.228 Restorable as boolean	534
* 4.36.229 screen as NSScreenMBS	534
* 4.36.230 sharingType as Integer	535
* 4.36.231 showsResizeIndicator as boolean	535
* 4.36.232 showsToolbarButton as boolean	535
* 4.36.233 styleMask as Integer	535
* 4.36.234 SubTitle as String	536
* 4.36.235 tabbingIdentifier as String	536
* 4.36.236 tabbingMode as Integer	537
* 4.36.237 Title as String	537
* 4.36.238 titlebarAppearsTransparent as Boolean	537
* 4.36.239 titlebarSeparatorStyle as Integer	538

	99
* 4.36.240 titleVisibility as Integer	538
* 4.36.242 toolbarStyle as Integer	538
* 4.36.243 Top as Double	539
* 4.36.244 userTabbingPreference as Integer	539
* 4.36.245 viewsNeedDisplay as boolean	539
* 4.36.246 Visible as boolean	539
* 4.36.247 Width as Double	540
* 4.36.248 windowController as NSWindowControllerMBS	540
* 4.36.249 windowNumber as Integer	540
* 4.36.250 worksWhenModal as boolean	540
* 4.36.251 frameAutosaveName as string	541

• 10 Cocoa Toolbar	1043
– 4.36.1 class NSWindowMBS	469
* 4.36.120 runToolbarCustomizationPalette	503
* 4.36.241 toolbar as Variant	538

	101
• 4 Cocoa	147
– 4.36.1 class NSWindowMBS	469
* 4.36.3 addChildWindow(win as DesktopWindow, order as integer)	470
* 4.36.4 addChildWindow(win as NSWindowMBS, order as Integer)	470
* 4.36.5 addChildWindow(win as window, order as Integer)	471
* 4.36.6 addTabbedWindow(win as NSWindowMBS, ordered as Integer)	471
* 4.36.7 animator as NSWindowMBS	472
* 4.36.8 areCursorRectsEnabled as boolean	472
* 4.36.9 attachedSheet as NSWindowMBS	472
* 4.36.10 autorecalculatesContentBorderThicknessForEdge(edge as Integer) as boolean	473
* 4.36.11 becomeKeyWindow	473
* 4.36.12 becomeMainWindow	473
* 4.36.13 cacheImageInRect(r as NSRectMBS)	473
* 4.36.14 Center	474
* 4.36.15 childWindows as NSWindowMBS()	474
* 4.36.16 ClearFocus	474
* 4.36.17 Close	474
* 4.36.18 Constructor(w as DesktopWindow)	475
* 4.36.19 Constructor(w as window)	475
* 4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false)	475
* 4.36.21 contentBorderThicknessForEdge(edge as Integer) as Double	476
* 4.36.22 contentRectForFrameRect(windowFrame as NSRectMBS) as NSRectMBS	477
* 4.36.23 contentRectForFrameRect(windowFrame as NSRectMBS, styleMask as UInt32) as NSRectMBS	477
* 4.36.24 convertBaseToScreen(p as NSPointMBS) as NSPointMBS	478
* 4.36.25 convertScreenToBase(p as NSPointMBS) as NSPointMBS	478
* 4.36.26 dataWithEPSInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.27 dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.28 deminiaturize	478
* 4.36.29 disableCursorRects	479
* 4.36.30 disableFlushWindow	479
* 4.36.31 disableScreenUpdatesUntilFlush	479
* 4.36.32 disableSnapshotRestoration	479
* 4.36.33 discardCachedImage	480
* 4.36.34 discardCursorRects	480
* 4.36.35 display	480
* 4.36.36 displayIfNeeded	480
* 4.36.37 dockTile as Variant	481
* 4.36.38 dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NS-SizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)	481

* 4.36.39 enableCursorRects	481
* 4.36.40 enableFlushWindow	481
* 4.36.41 enableSnapshotRestoration	482
* 4.36.42 endEditingFor(anObject as object = nil)	482
* 4.36.43 fieldEditor(createFlag as boolean = True, forObject as object = nil) as Variant	482
* 4.36.44 firstResponder as NSResponderMBS	483
* 4.36.45 flushWindow	484
* 4.36.46 flushWindowIfNeeded	484
* 4.36.47 frameRectForContentRect(windowContent as NSRectMBS) as NSRectMBS	484
* 4.36.48 frameRectForContentRect(windowContentRect as NSRectMBS, styleMask as UInt32) as NSRectMBS	485
* 4.36.49 GetFrame(byref left as Double, byref top as Double, byref width as Double, byref height as Double)	485
* 4.36.50 gState as Integer	485
* 4.36.51 Hide	485
* 4.36.52 inLiveResize as boolean	486
* 4.36.53 invalidateCursorRectsForView(View as NSViewMBS)	486
* 4.36.54 invalidateRestorableState	486
* 4.36.55 invalidateShadow	486
* 4.36.56 keyDown(e as NSEventMBS)	486
* 4.36.57 makeFirstResponder(r as NSResponderMBS) as boolean	486
* 4.36.58 makeKeyAndOrderFront	487
* 4.36.59 makeKeyWindow	488
* 4.36.60 makeMainWindow	488
* 4.36.61 mergeAllWindows	488
* 4.36.62 minFrameWidthWithTitle(WindowTitle as string, styleMask as UInt32) as Double	488
* 4.36.63 miniaturize	489
* 4.36.64 moveTabToNewWindow	489
* 4.36.65 NSDockWindowLevel as Integer	489
* 4.36.66 NSFloatingWindowLevel as Integer	489
* 4.36.67 NSMainMenuWindowLevel as Integer	489
* 4.36.68 NSModalPanelWindowLevel as Integer	490
* 4.36.69 NSNormalWindowLevel as Integer	490
* 4.36.70 NSPopUpMenuWindowLevel as Integer	490
* 4.36.71 NSScreenSaverWindowLevel as Integer	490
* 4.36.72 NSStatusWindowLevel as Integer	490
* 4.36.73 NSSubmenuWindowLevel as Integer	491
* 4.36.74 NSTornOffMenuWindowLevel as Integer	491
* 4.36.75 NSWindowDidBecomeKeyNotification as string	491
* 4.36.76 NSWindowDidBecomeMainNotification as string	491
* 4.36.77 NSWindowDidChangeScreenNotification as string	491

	103
* 4.36.78 NSWindowDidChangeScreenProfileNotification as string	492
* 4.36.79 NSWindowDidDeminiaturizeNotification as string	492
* 4.36.80 NSWindowDidEndLiveResizeNotification as string	492
* 4.36.81 NSWindowDidEndSheetNotification as string	493
* 4.36.82 NSWindowDidEnterFullScreenNotification as string	493
* 4.36.83 NSWindowDidEnterVersionBrowserNotification as string	493
* 4.36.84 NSWindowDidExitFullScreenNotification as string	494
* 4.36.85 NSWindowDidExitVersionBrowserNotification as string	494
* 4.36.86 NSWindowDidExposeNotification as string	494
* 4.36.87 NSWindowDidMiniaturizeNotification as string	494
* 4.36.88 NSWindowDidMoveNotification as string	495
* 4.36.89 NSWindowDidResignKeyNotification as string	495
* 4.36.90 NSWindowDidResignMainNotification as string	495
* 4.36.91 NSWindowDidResizeNotification as string	496
* 4.36.92 NSWindowDidUpdateNotification as string	496
* 4.36.93 NSWindowWillBeginSheetNotification as string	496
* 4.36.94 NSWindowWillCloseNotification as string	496
* 4.36.95 NSWindowWillEnterFullScreenNotification as string	497
* 4.36.96 NSWindowWillEnterVersionBrowserNotification as string	497
* 4.36.97 NSWindowWillExitFullScreenNotification as string	497
* 4.36.98 NSWindowWillExitVersionBrowserNotification as string	497
* 4.36.99 NSWindowWillMiniaturizeNotification as string	498
* 4.36.100 NSWindowWillMoveNotification as string	498
* 4.36.101 NSWindowWillStartLiveResizeNotification as string	498
* 4.36.102 orderBack	498
* 4.36.103 orderFront	499
* 4.36.104 orderFrontRegardless	499
* 4.36.105 orderOut	499
* 4.36.106 PerformClose	499
* 4.36.107 performMiniaturize	500
* 4.36.108 performWindowDragWithEvent(event as NSEventMBS)	500
* 4.36.109 performZoom	500
* 4.36.110 print	500
* 4.36.111 registerForDraggedTypes(Types() as string)	501
* 4.36.112 removeChildWindow(win as NSWindowMBS)	501
* 4.36.113 removeChildWindow(win as window)	501
* 4.36.114 removeFrameUsingName(name as string)	501
* 4.36.115 resetCursorRects	502
* 4.36.116 resignKeyWindow	502
* 4.36.117 resignMainWindow	502
* 4.36.118 resizeFlags as Integer	502
* 4.36.119 restoreCachedImage	503

* 4.36.121 saveFrameUsingName(s as String)	503
* 4.36.122 selectKeyViewFollowingView(view as NSViewMBS)	503
* 4.36.123 selectKeyViewPrecedingView(view as NSViewMBS)	504
* 4.36.124 selectNextKeyView	504
* 4.36.125 selectNextTab	504
* 4.36.126 selectPreviousKeyView	504
* 4.36.127 selectPreviousTab	505
* 4.36.128 sendEvent(e as NSEventMBS)	505
* 4.36.129 setAutorecalculatesContentBorderThickness(flag as boolean, edge as Integer)	505
* 4.36.130 setBottomCornerRounded(flag as boolean)	506
* 4.36.131 setContentBorderThickness(thickness as Double, edge as Integer)	506
* 4.36.132 setContentSize(size as NSSizeMBS)	507
* 4.36.133 setFrame(frameRect as NSRectMBS)	507
* 4.36.134 setFrame(frameRect as NSRectMBS, display as boolean)	507
* 4.36.135 setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)	508
* 4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double)	508
* 4.36.137 setFrameAutosaveName(name as String) as boolean	508
* 4.36.138 setFrameFromString(s as String)	509
* 4.36.139 setFrameOrigin(point as NSPointMBS)	509
* 4.36.140 setFrameTopLeftPoint(point as NSPointMBS)	509
* 4.36.141 setFrameUsingName(name as String, force as boolean = false) as boolean	510
* 4.36.142 setRestorationClass	510
* 4.36.143 setTitleWithRepresentedFile(filename as folderitem)	510
* 4.36.144 setTitleWithRepresentedFilename(filename as string)	510
* 4.36.145 Show	511
* 4.36.146 standardWindowButton(button as Integer) as Variant	511
* 4.36.147 standardWindowButton(button as Integer, StyleMask as Integer) as Variant	511
* 4.36.148 stringWithSavedFrame as String	512
* 4.36.149 tabbedWindows as NSWindowMBS()	512
* 4.36.150 toggleFullScreen	513
* 4.36.151 toggleTabBar	513
* 4.36.152 toggleToolbarShown	513
* 4.36.153 toolbarview as NSViewMBS	514
* 4.36.154 unregisterDraggedTypes	514
* 4.36.155 update	514
* 4.36.156 useOptimizedDrawing(value as boolean)	514
* 4.36.157 WindowHandle as Integer	515
* 4.36.158 windowNumberAtPoint(x as Double, y as Double, belowWindowWithWindowNumber as Integer = 0) as Integer	515
* 4.36.159 windowNumbersWithOptions(options as Integer = 0) as Integer()	515
* 4.36.160 zoom	516
* 4.36.162 acceptsMouseMovedEvents as boolean	516

	105
* 4.36.163 allowsAutomaticWindowTabbing as Boolean	516
* 4.36.164 allowsConcurrentViewDrawing as boolean	517
* 4.36.165 allowsToolTipsWhenApplicationIsInactive as boolean	517
* 4.36.166 alphaValue as Double	517
* 4.36.167 animationBehavior as Integer	518
* 4.36.168 aspectRatio as NSSizeMBS	518
* 4.36.169 Autodisplay as boolean	518
* 4.36.170 backgroundColor as NSColorMBS	519
* 4.36.171 backingLocation as Integer	519
* 4.36.172 backingScaleFactor as Double	519
* 4.36.173 backingType as Integer	520
* 4.36.174 canBecomeKeyWindow as boolean	520
* 4.36.175 canBecomeMainWindow as boolean	520
* 4.36.176 canBecomeVisibleWithoutLogin as boolean	521
* 4.36.177 canHide as boolean	521
* 4.36.178 canStoreColor as boolean	521
* 4.36.179 className as string	521
* 4.36.180 classPath as string	522
* 4.36.181 collectionBehavior as Integer	522
* 4.36.182 colorSpace as NSColorSpaceMBS	522
* 4.36.183 contentAspectRatio as NSSizeMBS	522
* 4.36.184 contentLayoutRect as NSRectMBS	522
* 4.36.185 contentMaxSize as NSSizeMBS	523
* 4.36.186 contentMinSize as NSSizeMBS	523
* 4.36.187 contentResizeIncrements as NSSizeMBS	523
* 4.36.188 contentView as NSViewMBS	523
* 4.36.189 currentEvent as NSEventMBS	523
* 4.36.190 deepestScreen as NSScreenMBS	524
* 4.36.191 depthLimit as Integer	524
* 4.36.192 displaysWhenScreenProfileChanges as boolean	524
* 4.36.193 frame as NSRectMBS	524
* 4.36.194 hasDynamicDepthLimit as boolean	525
* 4.36.195 hasShadow as boolean	525
* 4.36.196 Height as Double	525
* 4.36.197 hidesOnDeactivate as boolean	525
* 4.36.198 identifier as string	525
* 4.36.199 ignoresMouseEvents as boolean	526
* 4.36.200 initialFirstResponder as NSViewMBS	526
* 4.36.201 isDocumentEdited as boolean	526
* 4.36.202 isExcludedFromWindowsMenu as boolean	527
* 4.36.203 isFlushWindowDisabled as boolean	527
* 4.36.204 isKeyWindow as boolean	528

* 4.36.205 isMainWindow as boolean	528
* 4.36.206 isMiniaturized as boolean	528
* 4.36.207 isMovableByWindowBackground as boolean	528
* 4.36.208 isOnActiveSpace as boolean	529
* 4.36.209 isOneShot as boolean	529
* 4.36.210 isOpaque as boolean	529
* 4.36.211 isSheet as boolean	529
* 4.36.212 isZoomed as boolean	529
* 4.36.213 Left as Double	530
* 4.36.214 Level as Integer	530
* 4.36.215 maxSize as NSSizeMBS	530
* 4.36.216 miniwindowImage as Variant	530
* 4.36.217 miniwindowTitle as String	531
* 4.36.218 minSize as NSSizeMBS	531
* 4.36.219 Movable as boolean	531
* 4.36.220 parentWindow as NSWindowMBS	531
* 4.36.221 preferredBackingLocation as Integer	532
* 4.36.222 preservesContentDuringLiveResize as boolean	532
* 4.36.223 preventsApplicationTerminationWhenModal as boolean	532
* 4.36.224 representedFile as folderitem	532
* 4.36.225 representedFilename as string	533
* 4.36.226 representedURL as string	533
* 4.36.227 resizeIncrements as NSSizeMBS	534
* 4.36.228 Restorable as boolean	534
* 4.36.229 screen as NSScreenMBS	534
* 4.36.230 sharingType as Integer	535
* 4.36.231 showsResizeIndicator as boolean	535
* 4.36.232 showsToolbarButton as boolean	535
* 4.36.233 styleMask as Integer	535
* 4.36.234 SubTitle as String	536
* 4.36.235 tabbingIdentifier as String	536
* 4.36.236 tabbingMode as Integer	537
* 4.36.237 Title as String	537
* 4.36.238 titlebarAppearsTransparent as Boolean	537
* 4.36.239 titlebarSeparatorStyle as Integer	538
* 4.36.240 titleVisibility as Integer	538
* 4.36.242 toolbarStyle as Integer	538
* 4.36.243 Top as Double	539
* 4.36.244 userTabbingPreference as Integer	539
* 4.36.245 viewsNeedDisplay as boolean	539
* 4.36.246 Visible as boolean	539
* 4.36.247 Width as Double	540

	107
* 4.36.248 windowController as NSWindowControllerMBS	540
* 4.36.249 windowNumber as Integer	540
* 4.36.250 worksWhenModal as boolean	540
* 4.36.251 frameAutosaveName as string	541

• 10 Cocoa Toolbar	1043
– 4.36.1 class NSWindowMBS	469
* 4.36.120 runToolbarCustomizationPalette	503
* 4.36.241 toolbar as Variant	538

	109
• 4 Cocoa	147
– 4.36.1 class NSWindowMBS	469
* 4.36.3 addChildWindow(win as DesktopWindow, order as integer)	470
* 4.36.4 addChildWindow(win as NSWindowMBS, order as Integer)	470
* 4.36.5 addChildWindow(win as window, order as Integer)	471
* 4.36.6 addTabbedWindow(win as NSWindowMBS, ordered as Integer)	471
* 4.36.7 animator as NSWindowMBS	472
* 4.36.8 areCursorRectsEnabled as boolean	472
* 4.36.9 attachedSheet as NSWindowMBS	472
* 4.36.10 autorecalculatesContentBorderThicknessForEdge(edge as Integer) as boolean	473
* 4.36.11 becomeKeyWindow	473
* 4.36.12 becomeMainWindow	473
* 4.36.13 cacheImageInRect(r as NSRectMBS)	473
* 4.36.14 Center	474
* 4.36.15 childWindows as NSWindowMBS()	474
* 4.36.16 ClearFocus	474
* 4.36.17 Close	474
* 4.36.18 Constructor(w as DesktopWindow)	475
* 4.36.19 Constructor(w as window)	475
* 4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false)	475
* 4.36.21 contentBorderThicknessForEdge(edge as Integer) as Double	476
* 4.36.22 contentRectForFrameRect(windowFrame as NSRectMBS) as NSRectMBS	477
* 4.36.23 contentRectForFrameRect(windowFrame as NSRectMBS, styleMask as UInt32) as NSRectMBS	477
* 4.36.24 convertBaseToScreen(p as NSPointMBS) as NSPointMBS	478
* 4.36.25 convertScreenToBase(p as NSPointMBS) as NSPointMBS	478
* 4.36.26 dataWithEPSInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.27 dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock	478
* 4.36.28 deminiaturize	478
* 4.36.29 disableCursorRects	479
* 4.36.30 disableFlushWindow	479
* 4.36.31 disableScreenUpdatesUntilFlush	479
* 4.36.32 disableSnapshotRestoration	479
* 4.36.33 discardCachedImage	480
* 4.36.34 discardCursorRects	480
* 4.36.35 display	480
* 4.36.36 displayIfNeeded	480
* 4.36.37 dockTile as Variant	481
* 4.36.38 dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NS-SizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)	481

* 4.36.39 enableCursorRects	481
* 4.36.40 enableFlushWindow	481
* 4.36.41 enableSnapshotRestoration	482
* 4.36.42 endEditingFor(anObject as object = nil)	482
* 4.36.43 fieldEditor(createFlag as boolean = True, forObject as object = nil) as Variant	482
* 4.36.44 firstResponder as NSResponderMBS	483
* 4.36.45 flushWindow	484
* 4.36.46 flushWindowIfNeeded	484
* 4.36.47 frameRectForContentRect(windowContent as NSRectMBS) as NSRectMBS	484
* 4.36.48 frameRectForContentRect(windowContentRect as NSRectMBS, styleMask as UInt32) as NSRectMBS	485
* 4.36.49 GetFrame(byref left as Double, byref top as Double, byref width as Double, byref height as Double)	485
* 4.36.50 gState as Integer	485
* 4.36.51 Hide	485
* 4.36.52 inLiveResize as boolean	486
* 4.36.53 invalidateCursorRectsForView(View as NSViewMBS)	486
* 4.36.54 invalidateRestorableState	486
* 4.36.55 invalidateShadow	486
* 4.36.56 keyDown(e as NSEventMBS)	486
* 4.36.57 makeFirstResponder(r as NSResponderMBS) as boolean	486
* 4.36.58 makeKeyAndOrderFront	487
* 4.36.59 makeKeyWindow	488
* 4.36.60 makeMainWindow	488
* 4.36.61 mergeAllWindows	488
* 4.36.62 minFrameWidthWithTitle(WindowTitle as string, styleMask as UInt32) as Double	488
* 4.36.63 miniaturize	489
* 4.36.64 moveTabToNewWindow	489
* 4.36.65 NSDockWindowLevel as Integer	489
* 4.36.66 NSFloatingWindowLevel as Integer	489
* 4.36.67 NSMainMenuWindowLevel as Integer	489
* 4.36.68 NSModalPanelWindowLevel as Integer	490
* 4.36.69 NSNormalWindowLevel as Integer	490
* 4.36.70 NSPopUpMenuWindowLevel as Integer	490
* 4.36.71 NSScreenSaverWindowLevel as Integer	490
* 4.36.72 NSStatusWindowLevel as Integer	490
* 4.36.73 NSSubmenuWindowLevel as Integer	491
* 4.36.74 NSTornOffMenuWindowLevel as Integer	491
* 4.36.75 NSWindowDidBecomeKeyNotification as string	491
* 4.36.76 NSWindowDidBecomeMainNotification as string	491
* 4.36.77 NSWindowDidChangeScreenNotification as string	491

* 4.36.78	NSNotification as string	492
* 4.36.79	NSNotification as string	492
* 4.36.80	NSNotification as string	492
* 4.36.81	NSNotification as string	493
* 4.36.82	NSNotification as string	493
* 4.36.83	NSNotification as string	493
* 4.36.84	NSNotification as string	494
* 4.36.85	NSNotification as string	494
* 4.36.86	NSNotification as string	494
* 4.36.87	NSNotification as string	494
* 4.36.88	NSNotification as string	495
* 4.36.89	NSNotification as string	495
* 4.36.90	NSNotification as string	495
* 4.36.91	NSNotification as string	496
* 4.36.92	NSNotification as string	496
* 4.36.93	NSNotification as string	496
* 4.36.94	NSNotification as string	496
* 4.36.95	NSNotification as string	497
* 4.36.96	NSNotification as string	497
* 4.36.97	NSNotification as string	497
* 4.36.98	NSNotification as string	497
* 4.36.99	NSNotification as string	498
* 4.36.100	NSNotification as string	498
* 4.36.101	NSNotification as string	498
* 4.36.102	orderBack	498
* 4.36.103	orderFront	499
* 4.36.104	orderFrontRegardless	499
* 4.36.105	orderOut	499
* 4.36.106	performClose	499
* 4.36.107	performMiniaturize	500
* 4.36.108	performWindowDragWithEvent(event as NSEventMBS)	500
* 4.36.109	performZoom	500
* 4.36.110	print	500
* 4.36.111	registerForDraggedTypes(Types() as string)	501
* 4.36.112	removeChildWindow(win as NSWindowMBS)	501
* 4.36.113	removeChildWindow(win as window)	501
* 4.36.114	removeFrameUsingName(name as string)	501
* 4.36.115	resetCursorRects	502
* 4.36.116	resignKeyWindow	502
* 4.36.117	resignKeyWindow	502
* 4.36.118	resizeFlags as Integer	502
* 4.36.119	restoreCachedImage	503

* 4.36.121 saveFrameUsingName(s as String)	503
* 4.36.122 selectKeyViewFollowingView(view as NSViewMBS)	503
* 4.36.123 selectKeyViewPrecedingView(view as NSViewMBS)	504
* 4.36.124 selectNextKeyView	504
* 4.36.125 selectNextTab	504
* 4.36.126 selectPreviousKeyView	504
* 4.36.127 selectPreviousTab	505
* 4.36.128 sendEvent(e as NSEventMBS)	505
* 4.36.129 setAutorecalculatesContentBorderThickness(flag as boolean, edge as Integer)	505
* 4.36.130 setBottomCornerRounded(flag as boolean)	506
* 4.36.131 setContentBorderThickness(thickness as Double, edge as Integer)	506
* 4.36.132 setContentSize(size as NSSizeMBS)	507
* 4.36.133 setFrame(frameRect as NSRectMBS)	507
* 4.36.134 setFrame(frameRect as NSRectMBS, display as boolean)	507
* 4.36.135 setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)	508
* 4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double)	508
* 4.36.137 setFrameAutosaveName(name as String) as boolean	508
* 4.36.138 setFrameFromString(s as String)	509
* 4.36.139 setFrameOrigin(point as NSPointMBS)	509
* 4.36.140 setFrameTopLeftPoint(point as NSPointMBS)	509
* 4.36.141 setFrameUsingName(name as String, force as boolean = false) as boolean	510
* 4.36.142 setRestorationClass	510
* 4.36.143 setTitleWithRepresentedFile(filename as folderitem)	510
* 4.36.144 setTitleWithRepresentedFilename(filename as string)	510
* 4.36.145 Show	511
* 4.36.146 standardWindowButton(button as Integer) as Variant	511
* 4.36.147 standardWindowButton(button as Integer, StyleMask as Integer) as Variant	511
* 4.36.148 stringWithSavedFrame as String	512
* 4.36.149 tabbedWindows as NSWindowMBS()	512
* 4.36.150 toggleFullScreen	513
* 4.36.151 toggleTabBar	513
* 4.36.152 toggleToolbarShown	513
* 4.36.153 toolbarview as NSViewMBS	514
* 4.36.154 unregisterDraggedTypes	514
* 4.36.155 update	514
* 4.36.156 useOptimizedDrawing(value as boolean)	514
* 4.36.157 WindowHandle as Integer	515
* 4.36.158 windowNumberAtPoint(x as Double, y as Double, belowWindowWithWindowNumber as Integer = 0) as Integer	515
* 4.36.159 windowNumbersWithOptions(options as Integer = 0) as Integer()	515
* 4.36.160 zoom	516
* 4.36.162 acceptsMouseMovedEvents as boolean	516

	113
* 4.36.163 allowsAutomaticWindowTabbing as Boolean	516
* 4.36.164 allowsConcurrentViewDrawing as boolean	517
* 4.36.165 allowsToolTipsWhenApplicationIsInactive as boolean	517
* 4.36.166 alphaValue as Double	517
* 4.36.167 animationBehavior as Integer	518
* 4.36.168 aspectRatio as NSSizeMBS	518
* 4.36.169 Autodisplay as boolean	518
* 4.36.170 backgroundColor as NSColorMBS	519
* 4.36.171 backingLocation as Integer	519
* 4.36.172 backingScaleFactor as Double	519
* 4.36.173 backingType as Integer	520
* 4.36.174 canBecomeKeyWindow as boolean	520
* 4.36.175 canBecomeMainWindow as boolean	520
* 4.36.176 canBecomeVisibleWithoutLogin as boolean	521
* 4.36.177 canHide as boolean	521
* 4.36.178 canStoreColor as boolean	521
* 4.36.179 className as string	521
* 4.36.180 classPath as string	522
* 4.36.181 collectionBehavior as Integer	522
* 4.36.182 colorSpace as NSColorSpaceMBS	522
* 4.36.183 contentAspectRatio as NSSizeMBS	522
* 4.36.184 contentLayoutRect as NSRectMBS	522
* 4.36.185 contentMaxSize as NSSizeMBS	523
* 4.36.186 contentMinSize as NSSizeMBS	523
* 4.36.187 contentResizeIncrements as NSSizeMBS	523
* 4.36.188 contentView as NSViewMBS	523
* 4.36.189 currentEvent as NSEventMBS	523
* 4.36.190 deepestScreen as NSScreenMBS	524
* 4.36.191 depthLimit as Integer	524
* 4.36.192 displaysWhenScreenProfileChanges as boolean	524
* 4.36.193 frame as NSRectMBS	524
* 4.36.194 hasDynamicDepthLimit as boolean	525
* 4.36.195 hasShadow as boolean	525
* 4.36.196 Height as Double	525
* 4.36.197 hidesOnDeactivate as boolean	525
* 4.36.198 identifier as string	525
* 4.36.199 ignoresMouseEvents as boolean	526
* 4.36.200 initialFirstResponder as NSViewMBS	526
* 4.36.201 isDocumentEdited as boolean	526
* 4.36.202 isExcludedFromWindowsMenu as boolean	527
* 4.36.203 isFlushWindowDisabled as boolean	527
* 4.36.204 isKeyWindow as boolean	528

* 4.36.205 isMainWindow as boolean	528
* 4.36.206 isMiniaturized as boolean	528
* 4.36.207 isMovableByWindowBackground as boolean	528
* 4.36.208 isOnActiveSpace as boolean	529
* 4.36.209 isOneShot as boolean	529
* 4.36.210 isOpaque as boolean	529
* 4.36.211 isSheet as boolean	529
* 4.36.212 isZoomed as boolean	529
* 4.36.213 Left as Double	530
* 4.36.214 Level as Integer	530
* 4.36.215 maxSize as NSSizeMBS	530
* 4.36.216 miniwindowImage as Variant	530
* 4.36.217 miniwindowTitle as String	531
* 4.36.218 minSize as NSSizeMBS	531
* 4.36.219 Movable as boolean	531
* 4.36.220 parentWindow as NSWindowMBS	531
* 4.36.221 preferredBackingLocation as Integer	532
* 4.36.222 preservesContentDuringLiveResize as boolean	532
* 4.36.223 preventsApplicationTerminationWhenModal as boolean	532
* 4.36.224 representedFile as folderitem	532
* 4.36.225 representedFilename as string	533
* 4.36.226 representedURL as string	533
* 4.36.227 resizeIncrements as NSSizeMBS	534
* 4.36.228 Restorable as boolean	534
* 4.36.229 screen as NSScreenMBS	534
* 4.36.230 sharingType as Integer	535
* 4.36.231 showsResizeIndicator as boolean	535
* 4.36.232 showsToolbarButton as boolean	535
* 4.36.233 styleMask as Integer	535
* 4.36.234 SubTitle as String	536
* 4.36.235 tabbingIdentifier as String	536
* 4.36.236 tabbingMode as Integer	537
* 4.36.237 Title as String	537
* 4.36.238 titlebarAppearsTransparent as Boolean	537
* 4.36.239 titlebarSeparatorStyle as Integer	538
* 4.36.240 titleVisibility as Integer	538
* 4.36.242 toolbarStyle as Integer	538
* 4.36.243 Top as Double	539
* 4.36.244 userTabbingPreference as Integer	539
* 4.36.245 viewsNeedDisplay as boolean	539
* 4.36.246 Visible as boolean	539
* 4.36.247 Width as Double	540

	115
* 4.36.248 windowController as NSWindowControllerMBS	540
* 4.36.249 windowNumber as Integer	540
* 4.36.250 worksWhenModal as boolean	540
* 4.36.251 frameAutosaveName as string	541
– 4.37.1 class NSWindowRestoreHandlerMBS	545
* 4.37.3 Constructor	545
* 4.37.4 Destructor	545
* 4.37.5 SetLastError(error as NSErrorMBS)	545
* 4.37.6 SetRestoredWindow(win as DesktopWindow)	545
* 4.37.7 SetRestoredWindow(win as NSWindowMBS)	546
* 4.37.8 SetRestoredWindow(win as window)	546
* 4.37.10 RestoreWindow(identifier as string, state as Variant)	546

- **4 Cocoa** 147
 - 16.2.1 class Window 1131
 - * 16.2.3 NSPanelMBS as NSPanelMBS 1131
 - * 16.2.4 NSWindowMBS as NSWindowMBS 1131

Chapter 2

List of all classes

• ContainerControl	549
• Control	1045
• DesktopContainer	550
• DesktopControl	1046
• DesktopWindow	1129
• DRNotificationCenterMBS	1095
• NSAttributedStringMBS	925
• NSBezierPathMBS	551
• NSBitmapImageRepMBS	572
• NSBundleMBS	147
• NSCachedURLResponseMBS	797
• NSCalendarMBS	164
• NSCharacterSetMBS	173
• NSCoderMBS	186
• NSColorListMBS	587
• NSColorMBS	590
• NSColorSamplerMBS	640
• NSColorSpaceMBS	642
• NSCursorMBS	194

• NSDateComponentsMBS	209
• NSDirectoryEnumeratorMBS	215
• NSDistributedNotificationCenterMBS	1097
• NSEnumeratorMBS	225
• NSEventMBS	227
• NSFileWrapperMBS	984
• NSFontDescriptorMBS	249
• NSFontMBS	264
• NSHelpManagerMBS	282
• NSImageMBS	657
• NSImageRepMBS	747
• NSImageSymbolConfigurationMBS	752
• NSIndexSetMBS	285
• NSInputStreamMBS	295
• NSKeyedArchiverMBS	297
• NSKeyedUnarchiverMBS	301
• NSKeyValueObserverMBS	303
• NSLocaleDateMBS	1047
• NSLocaleMBS	1058
• NSLocaleNumberMBS	1082
• NSMenuItemMBS	763
• NSMenuMBS	779
• NSMutableAttributedStringMBS	1001
• NSMutableCharacterSetMBS	308
• NSMutableIndexSetMBS	311
• NSMutableURLRequestMBS	316
• NSNotificationCenterMBS	1102
• NSNotificationMBS	1106
• NSNotificationObserverMBS	1111

	119
• NSOutputStreamMBS	322
• NSPanelMBS	325
• NSPasteboardItemDataProviderMBS	121
• NSPasteboardItemMBS	123
• NSPasteboardMBS	127
• NSPDFImageRepMBS	759
• NSPICKImageRepMBS	761
• NSProcessInfoActivityMBS	1113
• NSProcessInfoMBS	1115
• NSResponderMBS	333
• NSScreenMBS	364
• NSShadowMBS	1012
• NSSoundDelegateMBS	372
• NSSoundMBS	373
• NSSStreamMBS	383
• NSTextAttachmentMBS	1015
• NSTextBlockMBS	1019
• NSTextListMBS	1025
• NSTextTableBlockMBS	1032
• NSTextTableMBS	1034
• NSTimeZoneMBS	386
• NSTouchMBS	393
• NSUndoManagerMBS	1036
• NSURLAuthenticationChallengeMBS	800
• NSURLCacheMBS	803
• NSURLConnectionFilterMBS	808
• NSURLCredentialMBS	820
• NSURLCredentialStorageMBS	824
• NSURLDownloadMBS	826

• NSURLMBS	836
• NSURLProtectionSpaceMBS	905
• NSURLRequestMBS	910
• NSURLResponseMBS	920
• NSUserDefaultsMBS	400
• NSUUIDMBS	419
• NSViewControllerMBS	423
• NSViewMBS	429
• NSViewTooltipMBS	463
• NSWindowControllerMBS	465
• NSWindowMBS	469
• NSWindowRestoreHandlerMBS	545
• Window	1131

Chapter 3

Clipboard

3.1 class NSPasteboardItemDataProviderMBS

3.1.1 class NSPasteboardItemDataProviderMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: This protocol is implemented by the data provider of a pasteboard item to provide the data for a particular UTI type.

Notes: You can specify an object as a pasteboard data provider for a pasteboard item using NSPasteboardItemMBS's setDataProviderForTypes method. The data provider must implement this protocol to provide data upon request.

Blog Entries

- [MBS Real Studio Plugins, version 13.1pr11](#)

3.1.2 Methods

3.1.3 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

3.1.4 Destructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The destructor.

3.1.5 Properties

3.1.6 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

3.1.7 Events

3.1.8 Finished(Pasteboard as NSPasteboardMBS)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Informs the receiver that the pasteboard no longer needs the data provider for any of its pasteboard items.

Notes: One data provider can provide data for more than one pasteboard item. This method is called when the pasteboard no longer needs the data provider for any of its pasteboard items. This can be either because the data provider has fulfilled all promises, or because ownership of the pasteboard has changed.

Available in OS X v10.6 and later.

3.1.9 provideDataForType(Pasteboard as NSPasteboardMBS, item as NSPasteboardItemMBS, type as string)

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Asks the receiver to provide data for a specified type to a given pasteboard. (required)

Notes: pasteboard: A pasteboard to which the receiver has promised to provide data.

item: A pasteboard item for which the receiver has promised to provide data

type: A UTI type string.

The receiver was previously set as the provider using setDataProviderForTypes.

Available in OS X v10.6 and later.

3.2 class NSPasteboardItemMBS

3.2.1 class NSPasteboardItemMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: NSPasteboardItem is a generic class to represent an item on a pasteboard.

Example:

```
dim n as new NSPasteboardItemMBS
```

```
dim type as string = NSPasteboardmbs.NSPasteboardTypeString
n.stringForType(type) = "Hello World"
MsgBox n.stringForType(type)
```

Notes: There are three main uses for an NSPasteboardItem object:

Providing data on the pasteboard.

You can create one or more pasteboard items, set data or data providers for types, and write to them pasteboard.

Customizing data already on the pasteboard.

As a delegate or subclass, you can retrieve the pasteboard items currently on the pasteboard, read the existing types and data and set new data and data providers for types as needed.

Retrieving data from the pasteboard.

You can retrieve pasteboard items from the pasteboard then read the data for types you're interested in.

A pasteboard item can be associated with a single pasteboard. When you create an item, it can be written to any pasteboard. When you pass an item to a pasteboard in `writeObjects`, that item becomes bound to the pasteboard it was written to. When you retrieve items from a pasteboard using `pasteboardItems`, the returned items are associated with the messaged pasteboard. Passing an item that is already associated with a pasteboard into `writeObjects`: causes an exception to be raised.

Pasteboard items are intended to be used during a single pasteboard interaction, not held onto and used repeatedly. A pasteboard item is only valid until the owner of the pasteboard changes.

Important: If a pasteboard item is stale because the pasteboard owner has changed, it returns `nil` or `false` values from its methods.

Blog Entries

- [MBS Real Studio Plugins, version 13.1pr11](#)

3.2.2 Methods

3.2.3 availableTypeFromArray(types() as string) as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns from a given array of types the the first type contained in the pasteboard item, according to the ordering of types.

Notes: types: An array of strings representing UTIs, arranged in order of preference (most preferred as the 0th element in the array).

Returns the first (according to the sender's ordering of types) type in types contained in the pasteboard item, or nil if the receiver does not contain any types given in types.

The method checks for UTI conformance of the requested types, preferring an exact match to conformance. Available in OS X v10.6 and later.

3.2.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

3.2.5 Destructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The destructor.

3.2.6 setDataProviderForType(dataProvider as NSPasteboardItemDataProviderMBS, types() as string) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the data provider for the specified types.

Notes: dataProvider: A pasteboard data provider.

types: An array of strings indicating the UTIs for the data representations dataProvider may provide.

Returns true if the data provider was set successfully, otherwise false.

This method registers the data provider to be messaged to provide the data for any of the specified types when requested.

Available in OS X v10.6 and later.

3.2.7 types as string()

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of UTI strings of the data types supported by the receiver.

Notes: Available in OS X v10.6 and later.

3.2.8 Properties

3.2.9 dataProvider as NSPasteboardItemDataProviderMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The last dataprovider used with setDataProviderForTypes.

Notes: You should keep your dataprovider alive, so the object is not destroyed while pasteboarditem is used.

In this property we keep a reference to the data provider for you.

(Read and Write property)

3.2.10 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

3.2.11 dataForType(type as string) as memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The value for a specified type as a memoryblock object.

Example:

```
dim n as new NSPasteboardItemMBS
```

```
dim data as MemoryBlock = "Hello World"
```

```
dim type as string = NSPasteboardmbs.NSPasteboardTypeString
```

```
n.dataForType(type) = data
```

```
dim d as MemoryBlock = n.dataForType(type)
MsgBox DefineEncoding(d, encodings.UTF8)
```

Notes: (Read and Write computed property)

3.2.12 propertyListForType(type as string) as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The value for a specified type as a property list.

Notes: Stores a property list object (array, dictionary, string, numbers, memoryblocks)

Available in OS X v10.6 and later.

(Read and Write computed property)

3.2.13 stringForType(type as string) as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The value for the specified type as a string.

Example:

```
dim n as new NSPasteboardItemMBS
```

```
dim type as string = NSPasteboardmbs.NSPasteboardTypeString
```

```
n.stringForType(type) = "Hello World"
```

```
MsgBox n.stringForType(type)
```

Notes: type: A UTI type string.

Available in OS X v10.6 and later.

(Read and Write computed property)

3.3 class NSPasteboardMBS

3.3.1 class NSPasteboardMBS

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class for the clipboard access on Cocoa.

Example:

```
dim p as new NSPasteboardMBS
```

```
MsgBox join(p.types,EndOfLine)
```

Notes: NSPasteboard objects transfer data to and from the pasteboard server. The server is shared by all running applications. It contains data that the user has cut or copied, as well as other data that one application wants to transfer to another. NSPasteboard objects are an application's sole interface to the server and to all pasteboard operations.

An NSPasteboard object is also used to transfer data between applications and service providers listed in each application's Services menu. The drag pasteboard (NSDragPboard) is used to transfer data that is being dragged by the user.

The plugin class NSPasteboardMBS does not implement all features of the NSPasteboard. If you miss something, please email us.

Blog Entries

- [MBS Xojo Plugins, version 21.4pr2](#)
- [Windows, Xojo and the Clipboard](#)
- [MBS Xojo Plugins, version 17.1pr3](#)
- [New MBS REALbasic Plugin Version 10.4](#)
- [MBS REALbasic Plugins Version 10.4 release notes](#)
- [MBS REALbasic Plugins, version 10.4pr12](#)
- [Teaser: Clipboard classes](#)

3.3.2 Methods

3.3.3 addType(type as string) as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds promises for the specified types to the first pasteboard item.

Notes: types: An array of strings, each of which specifies a type of data that can be provided to the pasteboard.

Returns the new change count, or 0 if there was an error adding the types.

This method adds promises for the specified types to the first pasteboard item.

You use this methods to declare additional types of data for the first pasteboard item in the receiver. You can also use it to replace existing types added by a previous `declareTypes` or `addTypes` call.

The `types` parameter specifies the types of data you are promising to the pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation). New types are added to the end of the list containing any existing types, if any.

If you specify a type that has already been declared, this method replaces the owner of that type with the value in `newOwner`. In addition, any data already already written to the pasteboard for that type is removed.

3.3.4 `addTypes(types() as string) as Integer`

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds promises for the specified types to the first pasteboard item.

Notes: types: An array of strings, each of which specifies a type of data that can be provided to the pasteboard.

Returns the new change count, or 0 if there was an error adding the types.

This method adds promises for the specified types to the first pasteboard item.

You use this methods to declare additional types of data for the first pasteboard item in the receiver. You can also use it to replace existing types added by a previous `declareTypes` or `addTypes` call.

The `types` parameter specifies the types of data you are promising to the pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation). New types are added to the end of the list containing any existing types, if any.

If you specify a type that has already been declared, this method replaces the owner of that type with the

value in newOwner. In addition, any data already already written to the pasteboard for that type is removed.

3.3.5 changeCount as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's change count.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSGeneralPboard)
```

```
MsgBox str(p.changeCount)
```

Notes: The change count starts at zero when a client creates the receiver and becomes the first owner. The change count subsequently increments each time the pasteboard ownership changes.

The change count is also returned from clearContents and declareTypes. You can therefore record the change count at the time that you take ownership of the pasteboard and later compare it with the value returned from changeCount to determine whether you still have ownership.

3.3.6 clearContents as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Clears the existing contents of the pasteboard.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSGeneralPboard)
```

```
MsgBox str(p.clearContents)
```

Notes: Clears the existing contents of the pasteboard, preparing it for new contents. This is the first step in providing data on the pasteboard.

Available in Mac OS X v10.6 and later.

3.3.7 Constructor

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new NSPasteboardMBS object linked to the general NSPasteboard object.

Example:

```
dim p as new NSPasteboardMBS
```

```
MsgBox join(p.types,EndOfLine)
```

Notes: On success the handle property is not 0.

See also:

- 3.3.8 Constructor(name as string) 130

3.3.8 Constructor(name as string)

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new NSPasteboardMBS object linked to the pasteboard with the specified name.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSFindPboard)
```

```
// shows last find string
```

```
MsgBox p.stringForType(p.NSPasteboardTypeString)
```

Notes: On success the handle property is not 0.

See also:

- 3.3.7 Constructor 129

3.3.9 declareType(type as string) as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Prepares the receiver for a change in its contents by declaring the new types of data it will contain.

Notes: type: An strings that specify the type of data that may be added to the new pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation).

Returns the receiver's new change count.

This method is the equivalent of invoking clearContents, implicitly writing the first pasteboard item, and then calling addTypes to promise types for the first pasteboard item.

In Mac OS X v10.5 and earlier, this method is the first step in writing data to the pasteboard and must precede the messages that actually write the data. A `declareTypes` message essentially changes the contents of the receiver: It invalidates the current contents of the receiver and increments its change count.

3.3.10 `declareTypes(types() as string) as Integer`

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Prepares the receiver for a change in its contents by declaring the new types of data it will contain.

Notes: `type`: An array of Strings that specify the types of data that may be added to the new pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation).

Returns the receiver's new change count.

This method is the equivalent of invoking `clearContents`, implicitly writing the first pasteboard item, and then calling `addTypes` to promise types for the first pasteboard item.

In Mac OS X v10.5 and earlier, this method is the first step in writing data to the pasteboard and must precede the messages that actually write the data. A `declareTypes` message essentially changes the contents of the receiver: It invalidates the current contents of the receiver and increments its change count.

3.3.11 `generalPasteboard as NSPasteboardMBS`

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the general NSPasteboard object.

Example:

```
dim p as NSPasteboardMBS = NSPasteboardMBS.generalPasteboard
MsgBox join(p.types,EndOfLine)
```

3.3.12 `name as string`

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's name.

Example:

```
dim p as new NSPasteboardMBS
```

```
MsgBox p.name
```

3.3.13 NSColorPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: NSColor data.

On Mac OS X v10.6 and later, use NSPasteboardTypeColor (and you read and write colors directly to and from the pasteboard).

3.3.14 NSDragPboard as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the names for pasteboards.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSDragPboard)
```

```
MsgBox p.name
```

Notes: The pasteboard that stores data to be moved as the result of a drag operation.

For additional information on working with the drag pasteboard, see Drag and Drop Programming Topics for Cocoa.

3.3.15 NSFileNamesPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: An array of strings designating one or more filenames.

On Mac OS X v10.6 and later, use writeObjects to write file URLs to the pasteboard.

3.3.16 NSFilesPromisePboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Promised files.

On Mac OS X v10.6 and later, use `kPasteboardTypeFileURLPromise` instead.

For information on promised files, see *Dragging Files in Drag and Drop Programming Topics for Cocoa*.

3.3.17 NSFindPboard as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the names for pasteboards.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSFindPboard)
MsgBox p.name
```

Notes: The pasteboard that holds information about the current state of the active application's find panel. This information permits users to enter a search string into the find panel, then switch to another application to conduct another search.

3.3.18 NSFontPboard as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the names for pasteboards.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSFontPboard)
MsgBox p.name
```

Notes: The pasteboard that holds font and character information and supports Copy Font and Paste Font commands that may be implemented in a text editor.

3.3.19 NSFontPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Font and character information.

On Mac OS X v10.6 and later, use `NSPasteboardTypeFont` instead.

3.3.20 NSGeneralPboard as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the names for pasteboards.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSGeneralPboard)
MsgBox p.name
```

Notes: The pasteboard that's used for ordinary cut, copy, and paste operations. This pasteboard holds the contents of the last selection that's been cut or copied.

3.3.21 NSHTMLPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: HTML (which an NSTextView object can read from, but not write to).

On Mac OS X v10.6 and later, use NSPasteboardTypeHTML instead.

3.3.22 NSInkTextPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Ink text data.

On Mac OS X v10.6 and later, use kUTTypeInkText instead.

For information on ink text objects, see Using Ink Services in Your Application.

3.3.23 NSMultipleTextSelectionPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Multiple text selection.

On Mac OS X v10.6 and later, use NSPasteboardTypeMultipleTextSelection instead.

Available in Mac OS X v10.5 and later.

3.3.24 NSPasteboardTypeColor as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Color data (an NSColor object).

Available in Mac OS X v10.6 and later.

3.3.25 NSPasteboardTypeFindPanelSearchOptions as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Type for the Find panel metadata property list.

Available in Mac OS X v10.6 and later.

3.3.26 NSPasteboardTypeFont as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Font and character information.

Available in Mac OS X v10.6 and later.

3.3.27 NSPasteboardTypeHTML as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: HTML data.

Available in Mac OS X v10.6 and later.

3.3.28 NSPasteboardTypeMultipleTextSelection as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Multiple text selection.

Available in Mac OS X v10.6 and later.

3.3.29 NSPasteboardTypePDF as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: PDF data.

Available in Mac OS X v10.6 and later.

3.3.30 NSPasteboardTypePNG as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: PNG image data.

Available in Mac OS X v10.6 and later.

3.3.31 NSPasteboardTypeRTF as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Rich Text Format (RTF) data.

Available in Mac OS X v10.6 and later.

3.3.32 NSPasteboardTypeRTFD as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: RTFD formatted file contents.

Available in Mac OS X v10.6 and later.

3.3.33 NSPasteboardTypeRuler as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Paragraph formatting information.

Available in Mac OS X v10.6 and later.

3.3.34 NSPasteboardTypeSound as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Sound data (an NSSound object).

Available in Mac OS X v10.6 and later.

3.3.35 NSPasteboardTypeString as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: String data.

Available in Mac OS X v10.6 and later.

3.3.36 NSPasteboardTypeTabularText as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: A string containing tab-separated fields of text.

Available in Mac OS X v10.6 and later.

3.3.37 NSPasteboardTypeTIFF as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the constants for defining UTIs for common pasteboard data types.

Notes: Tag Image File Format (TIFF) data.

Available in Mac OS X v10.6 and later.

3.3.38 NSPDFPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: PDF data.

On Mac OS X v10.6 and later, use NSPasteboardTypePDF instead.

3.3.39 NSPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: QuickDraw picture data.

The PICT format was formally deprecated in Mac OS X v10.4 along with QuickDraw. You should not be explicitly providing or looking for PICT data on the pasteboard.

To aid in this deprecation, if PICT is the only image type on the pasteboard, as is sometimes the case when copying images from 32-bit Carbon applications, a translated image type will be automatically reported and provided by NSPasteboard. The translated type is added to the types array ahead of PICT so that the deprecated PICT format is not the preferred format. In addition, when an application provides image data to NSPasteboard, the Carbon Pasteboard Manager will automatically make a PICT translation available to 32-bit Carbon applications.

Although NSPboardType, and its UTI equivalent kUTTypePICT, will appear in a pasteboard's type array retrieved from the existing NSPasteboard API, it may cease to be reported in future releases.

3.3.40 NSPostScriptPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Encapsulated PostScript (EPS) code.

On Mac OS X v10.6 and later, use @"com.adobe.encapsulated-postscript" instead.

3.3.41 NSRTFDPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: RTFD formatted file contents.

On Mac OS X v10.6 and later, use NSPasteboardTypeRTFD instead.

3.3.42 NSRTFPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Rich Text Format (RTF) data.

On Mac OS X v10.6 and later, use NSPasteboardTypeRTF instead.

3.3.43 NSRulerPboard as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the names for pasteboards.

Example:

```
dim p as new NSPasteboardMBS(NSPasteboardMBS.NSRulerPboard)
MsgBox p.name
```

Notes: The pasteboard that holds information about paragraph formats in support of the Copy Ruler and Paste Ruler commands that may be implemented in a text editor.

3.3.44 NSRulerPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Paragraph formatting information.

On Mac OS X v10.6 and later, use NSPasteboardTypeRuler instead.

3.3.45 NSStringPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: String data.

On Mac OS X v10.6 and later, use NSPasteboardTypeString instead.

3.3.46 NSTabularTextPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: A string containing tab-separated fields of text.

On Mac OS X v10.6 and later, use NSPasteboardTypeTabularText instead.

3.3.47 NSTIFFPboardType as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: Tag Image File Format (TIFF) data.

On Mac OS X v10.6 and later, use `NSPasteboardTypeTIFF` instead.

3.3.48 `NSURLPboardType` as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: NSURL data for one file or resource.

On Mac OS X v10.6 and later, use `writeObjects:` to write URLs directly to the pasteboard instead.

On Mac OS X v10.5 and earlier: to write an URL to a pasteboard you use `writeToPasteboard:` (NSURL); to get an URL from a pasteboard you use `URLFromPasteboard:` (NSURL).

3.3.49 `NSVCardPboardType` as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the common pasteboard data types.

Notes: VCard data.

On Mac OS X v10.6 and later, use `kUTTypeVCard` instead.

3.3.50 `pasteboardItems` as `NSPasteboardItemMBS()`

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns all the items held by the receiver.

Notes: Returns all the items held by the receiver, or nil if there is an error retrieving pasteboard items. Available in OS X v10.6 and later.

3.3.51 `pasteboardWithName(name as string)` as `NSPasteboardMBS`

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the pasteboard with the specified name.

Example:

```
dim p as NSPasteboardMBS = NSPasteboardMBS.pasteboardWithName(NSPasteboardMBS.NSFindPboard)
```

```
// shows last find string
```

```
MsgBox p.stringForType(p.NSPasteboardTypeString)
```

3.3.52 pasteboardWithUniqueName as NSPasteboardMBS

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns a new pasteboard with a name that is guaranteed to be unique with respect to other pasteboards on the computer.

Notes: This method is useful for applications that implement their own interprocess communication using pasteboards.

3.3.53 releaseGlobally

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Releases the receiver's resources in the pasteboard server.

Notes: After this method is invoked, no other application can use the receiver.

A temporary, privately named pasteboard can be released this way when it is no longer needed, but a standard pasteboard should never be released globally.

3.3.54 SetPasteboardItems(items() as NSPasteboardItemMBS) as Boolean

Plugin Version: 17.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes pasteboard items to the pasteboard.

Notes: Returns true if the array was successfully added, otherwise false.

See also:

- 3.3.55 SetPasteboardItems(items() as Variant) as boolean

141

3.3.55 SetPasteboardItems(items() as Variant) as boolean

Plugin Version: 21.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes pasteboard items to the pasteboard.

Example:

Dim paths() As String

```
paths.Append "/Users/cs/Pictures/test1.jpg"  
paths.Append "/Users/cs/Pictures/test2.png"
```

```

paths.Append "/Users/cs/Pictures/test3.jpg"

Dim n As NSPasteboardMBS = NSPasteboardMBS.generalPasteboard

Dim files() As Variant
For Each path As String In paths

files.Append new FolderItem(path, FolderItem.PathModes.Native)

Next

Call n.clearContents

If n.SetPasteboardItems(files) Then
Break // okay
Else
Break // failed
End If

```

Notes: This method allows you to mix NSPasteboardItemMBS, folderitem, NSURLMBS, strings, NSAttributedStringMBS and other data to put on clipboard.

Returns true if the array was successfully added, otherwise false.

See also:

- 3.3.54 SetPasteboardItems(items() as NSPasteboardItemMBS) as Boolean

141

3.3.56 types as string()

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of the receiver's supported data types.

Example:

```

dim p as new NSPasteboardMBS

MsgBox join(p.types,EndOfLine)

```

Notes: An array of Strings containing the union of the types of data declared for all the pasteboard items on the receiver. The returned types are listed in the order they were declared.

You must send a types or availableTypeFromArray message before reading any data from an NSPasteboard object.

3.3.57 URLFromPasteboard as string

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Reads the URL string off of pasteboard.

Notes: Returns "" if pasteboard does not contain data of type NSURLPboardType.

3.3.58 writeURLToPasteboard(URL as string)

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes the receiver to pasteboard.

Notes: You must declare an NSURLPboardType data type for pasteboard before invoking this method; otherwise it returns without doing anything.

3.3.59 Properties

3.3.60 Handle as Integer

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal reference to the NSPasteBoard object.

Example:

```
dim p as new NSPasteboardMBS
```

```
MsgBox hex(p.Handle)
```

Notes: (Read and Write property)

3.3.61 dataForType(type as string) as Memoryblock

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get the data for the specified type from the first item in the receiver that contains the type or sets the given data as the representation for the specified type for the first item on the receiver.

Example:

```
dim p as new NSPasteboardMBS
```

```
// may have chr(0) on the end!  
MsgBox p.dataForType(p.NSPasteboardTypeString)
```

Notes: Returns a data object containing the data for the specified type from the first item in the receiver that contains the type, or "" if the contents of the pasteboard changed since they were last checked.

This method may also return nil if the pasteboard server cannot supply the data in time—for example, if the pasteboard’s owner is slow in responding to a `pasteboard:provideDataForType:` message and the interprocess communication times out.

Errors other than a timeout raise a `NSPasteboardCommunicationException` (a `NSErrorMBS` in Xojo).

If "" is returned, the application should put up a panel informing the user that it was unable to carry out the paste operation.

For standard text data types such as string, RTF, and RTFD, the text data from each item is returned as one combined result separated by newlines.

(Read and Write computed property)

3.3.62 `propertyListForType(type as string)` as Variant

Plugin Version: 13.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The property list for the specified type from the first item in the receiver that contains the type.

Notes: `dataType:` The pasteboard data type containing the property-list data.

The property list for the specified type from the first item in the receiver that contains the type. This object consists of `NSArray`, `NSData`, `NSDictionary`, or `NSString` objects—or any combination thereof.

This method invokes the `dataForType` method.

You must send `types` or `availableTypeFromArray` before invoking `propertyListForType`.

(Read and Write computed property)

3.3.63 `stringForType(type as string)` as string

Plugin Version: 10.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Get/Set the given string as the representation for the specified type for the first item on the receiver.

Example:

```
dim p as new NSPasteboardMBS
```

MsgBox p.stringForType(p.NSPasteboardTypeString)

Notes: A concatenation of the strings for the specified type from all the items in the receiver that contain the type, or nil if none of the items contain strings of the specified type.

This method invokes `dataForType` to obtain the string. If the string cannot be obtained, `stringForType` returns `""`. See `dataForType` for a description of what will cause `""` to be returned.

In Mac OS X v10.6 and later, if the receiver contains multiple items that can provide string, RTF, or RTFD data, the text data from each item is returned as a combined result separated by newlines.
(Read and Write computed property)

Chapter 4

Cocoa

4.1 class NSBundleMBS

4.1.1 class NSBundleMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: An NSBundle object represents a location in the file system that groups code and resources that can be used in a program.

Example:

```
MsgBox NSBundleMBS.mainBundle.bundlePath
```

Notes: NSBundle objects locate program resources, dynamically load and unload executable code, and assist in localization. You build a bundle in Xcode using one of these project types: Application, Framework, plugins.

Although bundle structures vary depending on the target platform and the type of bundle you are building, the NSBundle class hides this underlying structure in most (but not all) cases. Many of the methods you use to load resources from a bundle automatically locate the appropriate starting directory and look for resources in known places.

Blog Entries

- [MBS Xojo Plugins, version 23.3pr2](#)
- [MBS Xojo Plugins, version 20.6pr1](#)
- [NSBundleMBS and NSDirectoryEnumeratorMBS](#)

4.1.2 Methods

4.1.3 allBundles as NSBundleMBS()

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an array of all the application's non-framework bundles.

Example:

```
dim a(-1) as NSBundleMBS = NSBundleMBS.allBundles
dim lines(-1) as string
```

```
for each n as NSBundleMBS in a
lines.Append n.bundlePath
next
```

```
MsgBox Join(lines,EndOfLine)
```

Notes: The returned array includes the main bundle and all bundles that have been dynamically created but doesn't contain any bundles that represent frameworks.

4.1.4 allFrameworks as NSBundleMBS()

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an array of all of the application's bundles that represent frameworks.

Example:

```
dim a(-1) as NSBundleMBS = NSBundleMBS.allFrameworks
dim lines(-1) as string
```

```
for each n as NSBundleMBS in a
lines.Append n.bundlePath
next
```

```
MsgBox Join(lines,EndOfLine)
```

Notes: Returns an array of all of the application's bundles that represent frameworks. Only frameworks with one or more Objective-C classes in them are included.

The returned array includes frameworks that are linked into an application when the application is built and bundles for frameworks that have been dynamically created.

4.1.5 builtInPlugInsFolder as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing plug-ins.

Example:

```
MsgBox NSBundleMBS.mainBundle.builtInPlugInsFolder.DisplayName
```

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.6 builtInPlugInsPath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing plug-ins.

Example:

```
MsgBox NSBundleMBS.mainBundle.builtInPlugInsPath
```

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.7 bundleFolder as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's bundle directory.

Example:

```
MsgBox NSBundleMBS.mainBundle.bundleFolder.DisplayName
```

4.1.8 bundleIdentifier as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's bundle identifier.

Example:

```
dim n as NSBundleMBS = NSBundleMBS mainBundle
MsgBox n.bundleIdentifier
```

Notes: Returns the receiver's bundle identifier, which is defined by the `CFBundleIdentifier` key in the bundle's information property list.

4.1.9 bundlePath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's bundle directory.

Example:

```
MsgBox NSBundleMBS mainBundle.bundlePath
```

4.1.10 bundleWithIdentifier(identifier as string) as NSBundleMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the previously created NSBundle instance that has the specified bundle identifier.

Example:

```
// create with path
dim x as NSBundleMBS = NSBundleMBS.bundleWithPath(SpecialFolder.Applications.Child("iTunes.app"))
MsgBox x.bundlePath

// once the bundle is known it will be in the allBundles array and be found with bundleWithIdentifier:
dim n as NSBundleMBS = NSBundleMBS.bundleWithIdentifier("com.apple.iTunes")
MsgBox n.bundlePath
```

Notes: The previously created NSBundle instance that has the bundle identifier identifier. Returns nil if the requested bundle is not found.

This method is typically used by frameworks and plug-ins to locate their own bundle at runtime. This method may be somewhat more efficient than trying to locate the bundle using the `bundleForClass` method.

4.1.11 bundleWithPath(path as folderitem) as NSBundleMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an NSBundle object that corresponds to the specified directory.

Example:

```
// create with path
dim x as NSBundleMBS = NSBundleMBS.bundleWithPath(SpecialFolder.Applications.Child("iTunes.app"))
MsgBox x.bundlePath
```

Notes: Returns the NSBundle object that corresponds to fullPath, or nil if fullPath does not identify an accessible bundle directory.

This method allocates and initializes the returned object if there is no existing NSBundle associated with fullPath, in which case it returns the existing object.

See also:

- 4.1.12 bundleWithPath(path as string) as NSBundleMBS 151

4.1.12 bundleWithPath(path as string) as NSBundleMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an NSBundle object that corresponds to the specified directory.

Example:

```
// create with path
dim x as NSBundleMBS = NSBundleMBS.bundleWithPath("/Applications/iTunes.app")
MsgBox x.bundlePath
```

Notes: Returns the NSBundle object that corresponds to fullPath, or nil if fullPath does not identify an accessible bundle directory.

This method allocates and initializes the returned object if there is no existing NSBundle associated with fullPath, in which case it returns the existing object.

See also:

- 4.1.11 bundleWithPath(path as folderitem) as NSBundleMBS 150

4.1.13 Constructor(path as folderitem)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates an NSBundle object that corresponds to the specified directory.

Example:

```
dim n as new NSBundleMBS(SpecialFolder.Applications.Child("iTunes.app"))
dim d as Dictionary = n.infoDictionary
MsgBox d.Value("CFBundleName")
```

Notes: Creates the NSBundle object that corresponds to fullPath, or fails if fullPath does not identify an accessible bundle directory. On failure the handle property is 0.

This method allocates and initializes the returned object if there is no existing NSBundle associated with fullPath, in which case it returns the existing object.

See also:

- 4.1.14 Constructor(path as string) 152

4.1.14 Constructor(path as string)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates an NSBundle object that corresponds to the specified directory.

Example:

```
dim n as new NSBundleMBS("/Applications/iTunes.app")
dim d as Dictionary = n.infoDictionary
MsgBox d.Value("CFBundleName")
```

Notes: Creates the NSBundle object that corresponds to fullPath, or fails if fullPath does not identify an accessible bundle directory. On failure the handle property is 0.

This method allocates and initializes the returned object if there is no existing NSBundle associated with fullPath, in which case it returns the existing object.

See also:

- 4.1.13 Constructor(path as folderitem) 151

4.1.15 developmentLocalization as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the localization used to create the bundle.

Example:

```
MsgBox NSBundleMBS.mainBundle.developmentLocalization
```

Notes: The returned localization corresponds to the value in the `CFBundleDevelopmentRegion` key of the bundle's property list (Info.plist).

4.1.16 `executableArchitectures` as `Integer()`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an array of numbers indicating the architecture types supported by the bundle's executable.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.app")
dim b as new NSBundleMBS(f)

dim lines(-1) as string

for each e as Integer in b.executableArchitectures
dim s as string
Select case e
case NSBundleMBS.NSBundleExecutableArchitectureI386
s="i386"
case NSBundleMBS.NSBundleExecutableArchitecturePPC
s="PPC"
case NSBundleMBS.NSBundleExecutableArchitecturePPC64
s="PPC 64-bit"
case NSBundleMBS.NSBundleExecutableArchitectureX86_64
s="x86 64-bit"
case 12
s="iPhone"
else
s=Str(e)
end Select
lines.Append hex(e)+" ": "+s
next

MsgBox Join(lines,EndOfLine)
```

Notes: This method scans the bundle's Mach-O executable and returns all of the architecture types it finds. Because they are taken directly from the executable, the returned values may not always correspond to one of the well-known CPU types defined in "Mach-O Architecture."

4.1.17 executableFile as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's executable file.

Example:

```
MsgBox NSBundleMBS.mainBundle.executableFile.NativePath
```

4.1.18 executablePath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's executable file.

Example:

```
MsgBox NSBundleMBS.mainBundle.executablePath
```

4.1.19 infoDictionary as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns a dictionary that contains information about the receiver.

Example:

```
dim n as new NSBundleMBS(SpecialFolder.Applications.Child("iTunes.app"))
```

```
dim d as Dictionary = n.infoDictionary
```

```
dim lines(-1) as string
```

```
for each key as Variant in d.keys
```

```
dim value as Variant = d.Value(key)
```

```
// special handle folderitems
```

```
if value isa FolderItem then
```

```
dim f as FolderItem = value
```

```
value = f.Name
```

```
end if
```

```
lines.Append key.StringValue + " ->" + value.StringValue
```

```
next
```

```
MsgBox Join(lines,EndOfLine)
```

Notes: Returns a dictionary, constructed from the bundle’s Info.plist file, that contains information about the receiver. If the bundle does not contain an Info.plist file, a valid dictionary is returned but this dictionary contains only private keys that are used internally by the NSBundle class. The NSBundle class may add extra keys to this dictionary for its own use.

4.1.20 isLoaded as boolean

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Obtains information about the load status of a bundle.

Example:

```
MsgBox str(NSBundleMBS.mainBundle.isLoaded) // mainBundle is always loaded
```

Notes: Returns true if the bundle’s code is currently loaded, otherwise false.

4.1.21 load as boolean

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Dynamically loads the bundle’s executable code into a running program, if the code has not already been loaded.

Notes: Returns true if the method successfully loads the bundle’s code or if the code has already been loaded, otherwise false.

You can use this method to load the code associated with a dynamically loaded bundle, such as a plug-in or framework. Prior to Mac OS X version 10.5, a bundle would attempt to load its code—if it had any—only once. Once loaded, you could not unload that code. In Mac OS X version 10.5 and later, you can unload a bundle’s executable code using the unload method.

You don’t need to load a bundle’s executable code to search the bundle’s resources.

4.1.22 localizations as string()

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns a list of all the localizations contained within the receiver’s bundle.

Example:

```
MsgBox join(NSBundleMBS.mainBundle.localizations,EndOfLine)
```

Notes: Returns an array, containing strings, that specifies all the localizations contained within the receiver's bundle.

4.1.23 localizedInfoDictionary as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns a dictionary with the keys from the bundle's localized property list.

Example:

```
dim n as new NSBundleMBS(SpecialFolder.Applications.Child("iTunes.app"))
dim d as Dictionary = n.localizedInfoDictionary
dim lines(-1) as string

for each key as Variant in d.keys
dim value as Variant = d.Value(key)

// special handle folderitems
if value isa FolderItem then
dim f as FolderItem = value
value = f.Name
end if

lines.Append key.StringValue + " ->" + value.StringValue
next

MsgBox Join(lines,EndOfLine)
```

Notes: Returns a dictionary with the keys from the bundle's localized property list (InfoPlist.strings).

This method uses the preferred localization for the current user when determining which resources to return. If the preferred localization is not available, this method chooses the most appropriate localization found in the bundle.

4.1.24 localizedStringForKey(key as string, value as string="", tableName as string="") as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns a localized version of the string designated by the specified key and residing in the specified table.

Notes: key: The key for a string in the table identified by tableName.

value: The value to return if key is "" or if a localized string for key can't be found in the table.

tableName: The receiver's string table to search. If tableName is an empty string, the method attempts to use the table in Localizable.strings.

Returns a localized version of the string designated by key in table tableName. If value is nil or an empty string, and a localized string is not found in the table, returns key. If key and value are both nil, returns the empty string.

For more details about string localization and the specification of a .strings file, see "Working With Localized Strings."

Using the user default NSShowNonLocalizedStrings, you can alter the behavior of localizedStringForKey to log a message when the method can't find a localized string. If you set this default to true (in the global domain or in the application's domain), then when the method can't find a localized string in the table, it logs a message to the console and capitalizes key before returning it.

The following example cycles through a static array of keys when a button is clicked, gets the value for each key from a strings table named Buttons.strings, and sets the button title with the returned value:

4.1.25 mainBundle as NSBundleMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the NSBundle object that corresponds to the directory where the current application executable is located.

Example:

```
MsgBox NSBundleMBS.mainBundle.bundlePath
```

Notes: Returns the NSBundle object that corresponds to the directory where the application executable is located, or nil if a bundle object could not be created.

This method allocates and initializes a bundle object if one doesn't already exist. The new object corresponds to the directory where the application executable is located. Be sure to check the return value to make sure you have a valid bundle. This method may return a valid bundle object even for unbundled applications.

In general, the main bundle corresponds to an application file package or application wrapper: a directory that bears the name of the application and is marked by a ".app" extension.

4.1.26 pathForResource(name as string) as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the location of the specified image resource file.

Notes: name: The name of the image resource file, without any pathname information. Including a filename

extension is optional.

Returns the absolute pathname of the resource file or nil if the file was not found.

Image resources are those files in the bundle that are recognized by the `NSImage` class, including those that can be converted using the Image IO framework.

4.1.27 `pathForResource(name as string, extension as string) as folderitem`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname for the resource identified by the specified name and file extension.

Notes: name: The name of the resource file.

extension: If extension is an empty string, the extension is assumed not to exist and the file URL is the first file encountered that exactly matches name.

Returns the full pathname for the resource file or nil if the file could not be located.

The method first looks for a matching resource file in the non-localized resource directory of the specified bundle. (In Mac OS X, this directory is typically called Resources but in iPhone OS, it is the main bundle directory.) If a matching resource file is not found, it then looks in the top level of any available language-specific ".lproj" directories. (The search order for the language-specific directories corresponds to the user's preferences.) It does not recurse through other subdirectories at any of these locations. For more details see Bundles and Localization.

The following code fragment gets the path to a plist within the bundle, and loads it into a dictionary.

See also:

- 4.1.28 `pathForResource(name as string, extension as string, subpath as string) as folderitem` 158
- 4.1.29 `pathForResource(name as string, extension as string, subpath as string, localizationName as string) as folderitem` 159

4.1.28 `pathForResource(name as string, extension as string, subpath as string) as folderitem`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname for the resource identified by the specified name and file extension and located in the specified bundle subdirectory.

Notes: name: The name of the resource file.

extension: If extension is an empty string, the extension is assumed not to exist and the file URL is the first file encountered that exactly matches name.

subpath: The name of the bundle subdirectory.

Returns the full pathname for the resource file or nil if the file could not be located.

If subpath is "", this method searches the top-level nonlocalized resource directory and the top-level of any language-specific directories. (In Mac OS X, the top-level nonlocalized resource directory is typically called Resources but in iPhone OS, it is the main bundle directory.) For example, suppose you have a Mac OS X application with a modern bundle and you specify "Documentation" for the subpath parameter. This method would first look in the Contents/Resources/Documentation directory of the bundle, followed by the Documentation subdirectories of each language-specific .lproj directory. (The search order for the language-specific directories corresponds to the user's preferences.) This method does not recurse through any other subdirectories at any of these locations. For more details see Bundles and Localization.

See also:

- 4.1.27 pathForResource(name as string, extension as string) as folderitem 158
- 4.1.29 pathForResource(name as string, extension as string, subpath as string, localizationName as string) as folderitem 159

4.1.29 pathForResource(name as string, extension as string, subpath as string, localizationName as string) as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname for the resource identified by the specified name and file extension, located in the specified bundle subdirectory, and limited to global resources and those associated with the specified localization.

Notes: name: The name of the resource file.

extension: If extension is an empty string, the extension is assumed not to exist and the file URL is the first file encountered that exactly matches name.

subpath: The name of the bundle subdirectory to search.

localizationName: The name of the localization. This parameter should correspond to the name of one of the bundle's language-specific resource directories without the .lproj extension.

Returns the full pathname for the resource file or nil if the file could not be located.

This method is equivalent to pathForResource ofType, except that only nonlocalized resources and those in the language-specific .lproj directory specified by localizationName are searched.

There should typically be little reason to use this method—see Getting the Current Language and Locale. See also:

- 4.1.27 pathForResource(name as string, extension as string) as folderitem 158
- 4.1.28 pathForResource(name as string, extension as string, subpath as string) as folderitem 158

4.1.30 `pathForResource(name as string) as folderitem`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the location of the specified sound resource file.

Notes: `name`: The name of the sound resource file, without any pathname information. Including a filename extension is optional

Returns the folderitem of the resource file or nil if the file was not found.

Sound resources are those files in the bundle that are recognized by the `NSSound` class. The types of sound files can be determined by calling the `soundUnfilteredFileTypes` method of `NSSound`.

4.1.31 `preferredLocalizations as string()`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an array of strings indicating the actual localizations contained in the receiver's bundle.

Notes: An array of strings, each of which identifies the a localization in the receiver's bundle. The localizations in the array are not returned in any particular order.

4.1.32 `privateFrameworksFolder as folderitem`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing frameworks.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.33 `privateFrameworksPath as string`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing frameworks.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.34 `resourceFolder as folderitem`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiving bundle's subdirectory containing resources.

4.1.35 resourcePath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiving bundle's subdirectory containing resources.

4.1.36 sharedFrameworksFolder as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing shared frameworks.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.37 sharedFrameworksPath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing shared frameworks.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.38 sharedSupportFolder as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing shared support files.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.39 sharedSupportPath as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the full pathname of the receiver's subdirectory containing shared support files.

Notes: This method returns the appropriate path for modern application and framework bundles. This method may not return a path for non-standard bundle formats or for some older bundle formats.

4.1.40 unload as boolean

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Unloads the code associated with the receiver.

Notes: Returns true if the bundle was successfully unloaded or was not already loaded; otherwise, false if the bundle could not be unloaded.

This method attempts to unload a bundle's executable code using the underlying dynamic loader (typically dyld). You may use this method to unload plug-in and framework bundles when you no longer need the code they contain. You should use this method to unload bundles that were loaded using the methods of the NSBundle class only. Do not use this method to unload bundles that were originally loaded using the bundle-manipulation functions in Core Foundation.

It is the responsibility of the caller to ensure that no in-memory objects or data structures refer to the code being unloaded. For example, if you have an object whose class is defined in a bundle, you must release that object prior to unloading the bundle. Similarly, your code should not attempt to access any symbols defined in an unloaded bundle.

Prior to Mac OS X version 10.5, code could not be unloaded once loaded, and this method would always return false. In Mac OS X version 10.5 and later, you can unload a bundle's executable code using this method.

Available in Mac OS X v10.5 and later.

4.1.41 Properties

4.1.42 Handle as Integer

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The internal reference to the NSBundle object.

Example:

```
MsgBox hex(NSBundleMBS.mainBundle.Handle)
```

Notes: (Read and Write property)

4.1.43 Constants

Architectures

Constant	Value	Description
NSBundleExecutableArchitectureARM64	&h0100000c	Specifies the 64-bit ARM architecture.
NSBundleExecutableArchitectureI386	7	Specifies the 32-bit Intel architecture. Available in Mac OS X v10.5 and later.
NSBundleExecutableArchitecturePPC	&h12	Specifies the 32-bit PowerPC architecture. Available in Mac OS X v10.5 and later.
NSBundleExecutableArchitecturePPC64	&h01000012	Specifies the 64-bit PowerPC architecture. Available in Mac OS X v10.5 and later.
NSBundleExecutableArchitectureX86_64	&h01000007	Specifies the 64-bit Intel architecture. Available in Mac OS X v10.5 and later.

4.2 class NSCalendarMBS

4.2.1 class NSCalendarMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The calendar class for Cocoa.

Notes: see also

https://developer.apple.com/library/mac/#documentation/Cocoa/Reference/Foundation/Classes/NSCalendar_Class/Reference/NSCalendar.html

Blog Entries

- [News from the MBS Xojo Plugins Version 22.5](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.5](#)
- [MBS Xojo Plugins, version 22.5pr4](#)
- [News from the MBS Xojo Plugins Version 20.1](#)
- [MBS Xojo Plugins, version 20.1pr1](#)
- [MBS Real Studio Plugins, version 12.3pr11](#)

4.2.2 Methods

4.2.3 AMSymbol as string

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: The symbol used to represent ,ÁúAM,Àù for this calendar.

4.2.4 autoupdatingCurrentCalendar as NSCalendarMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the current logical calendar for the current user.

Notes: Settings you get from this calendar do change as the user's settings change (contrast with current-Calendar).

Note that if you cache values based on the calendar or related information those caches will of course not be automatically updated by the updating of the calendar object.

Available in Mac OS X v10.5 and later.

4.2.5 calendarIdentifier as string

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the identifier for the receiver.

4.2.6 calendarWithIdentifier(identifier as String) as NSCalendarMBS

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Creates a new calendar specified by a given identifier.

Notes: identifier: The identifier for the new calendar. For valid identifiers, see Calendar Identifiers.

Returns the initialized calendar, or nil if the identifier is unknown (if, for example, it is either an unrecognized string or the calendar is not supported by the current version of the operating system).

The returned calendar defaults to the current locale and default time zone.

4.2.7 componentsInTimeZone(timezone as NSTimeZoneMBS, date as Date) as NSDateComponentsMBS

Plugin Version: 20.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns all the date components of a date, as if in a given time zone (instead of the receiving calendar's time zone).

Notes: timezone: The time zone to use when returning the components. This value overrides the time zone of the receiving NSCalendar.

date: The date for which to perform the calculation.

Returns an NSDateComponents object containing all the components from the given date, calculated using the given time zone.

See also:

- 4.2.8 componentsInTimeZone(timezone as NSTimeZoneMBS, date as DateTime) as NSDateComponentsMBS 165

4.2.8 componentsInTimeZone(timezone as NSTimeZoneMBS, date as DateTime) as NSDateComponentsMBS

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Returns all the date components of a date, as if in a given time zone (instead of the receiving

calendar,Ãs time zone).

Notes: `timezone`: The time zone to use when returning the components. This value overrides the time zone of the receiving `NSCalendar`.

`date`: The date for which to perform the calculation.

Returns an `NSDateComponents` object containing all the components from the given date, calculated using the given time zone.

See also:

- 4.2.7 `componentsInTimeZone(timezone as NSTimeZoneMBS, date as Date) as NSDateComponentsMBS` 165

4.2.9 Constructor

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Creates new calendar object with current calendar.

See also:

- 4.2.10 `Constructor(identifier as string)` 166

4.2.10 Constructor(identifier as string)

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Initializes a newly-allocated `NSCalendar` object for the calendar specified by a given identifier.

See also:

- 4.2.9 `Constructor` 166

4.2.11 copy as NSCalendarMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns a copy of the calendar object.

4.2.12 currentCalendar as NSCalendarMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the logical calendar for the current user.

Notes: The returned calendar is formed from the settings for the current user's chosen system locale overlaid with any custom settings the user has specified in System Preferences. Settings you get from this calendar

do not change as System Preferences are changed, so that your operations are consistent (contrast with `autoUpdatingCurrentCalendar`).

4.2.13 `dateByAddingComponents(components as NSDateComponentsMBS, toDate as Date, Options as Integer = 0) as Date`

Plugin Version: 20.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a date representing the absolute time calculated by adding given components to a given date.

Notes: `components`: The components to add to date.

`date`: The date to which comps are added.

`options`: Options for the calculation. See `NSCalendarOptions` in Apple documentation for possible values.

If you specify no options, overflow in a unit carries into the higher units (as in typical addition).

Returns a new date representing the absolute time calculated by adding to date the calendrical components specified by comps using the options specified by opts. Returns nil if date falls outside the defined range of the receiver or if the computation cannot be performed.

4.2.14 `dateFromComponents(components as NSDateComponentsMBS) as Date`

Plugin Version: 20.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a date representing the absolute time calculated from given components.

Notes: `components`: The components from which to calculate the returned date.

Returns a new Date object representing the absolute time calculated from comps. Returns nil if the receiver cannot convert the components given in comps into an NSDate object.

4.2.15 `dateTimeByAddingComponents(components as NSDateComponentsMBS, toDate as DateTime, Options as Integer = 0) as DateTime`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Returns a date representing the absolute time calculated by adding given components to a given date.

Notes: `components`: The components to add to date.

`date`: The date to which comps are added.

`options`: Options for the calculation. See `NSCalendarOptions` in Apple documentation for possible values.

If you specify no options, overflow in a unit carries into the higher units (as in typical addition).

Returns a new date representing the absolute time calculated by adding to date the calendrical components specified by comps using the options specified by opts. Returns nil if date falls outside the defined range of the receiver or if the computation cannot be performed.

4.2.16 `dateTimeFromComponents(components as NSDateComponentsMBS) as DateTime`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Returns a date representing the absolute time calculated from given components.

Notes: components: The components from which to calculate the returned date.

Returns a new Date object representing the absolute time calculated from comps. Returns nil if the receiver cannot convert the components given in comps into an NSDate object.

4.2.17 `description as string`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The description for this timezone.

4.2.18 `NSCalendarIdentifierBuddhist as String`

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Buddhist calendar.

4.2.19 `NSCalendarIdentifierChinese as String`

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Chinese calendar.

4.2.20 `NSCalendarIdentifierCoptic as String`

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Coptic calendar.

4.2.21 NSCalendarIdentifierEthiopicAmeteAlem as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Ethiopic (Amete Alem) calendar.

4.2.22 NSCalendarIdentifierEthiopicAmeteMihret as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Ethiopic (Amete Mihret) calendar.

4.2.23 NSCalendarIdentifierGregorian as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Gregorian calendar.

4.2.24 NSCalendarIdentifierHebrew as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Hebrew calendar.

4.2.25 NSCalendarIdentifierIndian as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Indian calendar.

4.2.26 NSCalendarIdentifierIslamic as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Islamic calendar.

4.2.27 NSCalendarIdentifierIslamicCivil as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Islamic civil calendar.

4.2.28 NSCalendarIdentifierIslamicTabular as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for a tabular Islamic calendar.

4.2.29 NSCalendarIdentifierIslamicUmmAlQura as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Islamic Umm al-Qura calendar.

4.2.30 NSCalendarIdentifierISO8601 as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the ISO8601 calendar.

4.2.31 NSCalendarIdentifierJapanese as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Japanese calendar.

4.2.32 NSCalendarIdentifierPersian as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Persian calendar.

4.2.33 NSCalendarIdentifierRepublicOfChina as String

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: Identifier for the Republic of China calendar.

4.2.34 PMSymbol as string

Plugin Version: 22.5, Platform: macOS, Targets: All.

Function: The symbol used to represent ,ÁúPM,Àù for this calendar.

4.2.35 Print

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Writes the calendar to debug output.

Notes: This may help for debugging and you see output in console app.

4.2.36 Properties

4.2.37 Handle as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.2.38 firstWeekday as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The index of the first weekday.

Notes: (Read and Write computed property)

4.2.39 locale as NSLocaleMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The locale.

Notes: (Read and Write computed property)

4.2.40 `minimumDaysInFirstWeek` as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The minimum number of days in the first week.

Notes: (Read and Write computed property)

4.2.41 `timeZone` as `NSTimeZoneMBS`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The time zone.

Notes: (Read and Write computed property)

4.2.42 Constants

Calendar Units

Constant	Value	Description
<code>kCFCalendarUnitDay</code>	16	Specifies the day unit.
<code>kCFCalendarUnitEra</code>	2	Specifies the era unit.
<code>kCFCalendarUnitHour</code>	32	Specifies the hour unit.
<code>kCFCalendarUnitMinute</code>	64	Specifies the minute unit.
<code>kCFCalendarUnitMonth</code>	8	Specifies the month unit.
<code>kCFCalendarUnitQuarter</code>	2048	Specifies the quarter-year unit.
<code>kCFCalendarUnitSecond</code>	128	Specifies the second unit.
<code>kCFCalendarUnitWeekday</code>	256	Specifies the weekday unit. The weekday units are the numbers 1-N (where for the Gregorian calendar N=7 and 1 is Sunday).
<code>kCFCalendarUnitWeekdayOrdinal</code>	512	Specifies the ordinal weekday unit. The weekday ordinal unit describes ordinal position within the month using the corresponding weekday unit. For example, in the Gregorian calendar a weekday ordinal unit of 2 for a weekday unit 3 indicates "the second Tuesday in the month".
<code>kCFCalendarUnitWeekOfMonth</code>	4096	Specifies the original week of a month calendar unit.
<code>kCFCalendarUnitWeekOfYear</code>	8192	Specifies the original week of the year calendar unit.
<code>kCFCalendarUnitYear</code>	4)	Specifies the year unit.
<code>kCFCalendarUnitYearForWeekOfYear</code>	16384	Specifies the relative year for a week within a year calendar unit.

4.3 class NSCharacterSetMBS

4.3.1 class NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: An NSCharacterSet object represents a set of Unicode-compliant characters.

Example:

```
// get the Symbol font
dim n as NSFontMBS = NSFontMBS.fontWithName("Symbol",10)

// what characters are defined for this font?
dim c as NSCharacterSetMBS = n.coveredCharacterSet

// display a string with all the characters
MsgBox c.StringValue
```

Notes: NSString and NSScanner objects use NSCharacterSet objects to group characters together for searching operations, so that they can find any of a particular set of characters during a search. The cluster's two public classes, NSCharacterSet and NSMutableCharacterSet, declare the programmatic interface for static and dynamic character sets, respectively.

The objects you create using these classes are referred to as character set objects (and when no confusion will result, merely as character sets). Because of the nature of class clusters, character set objects aren't actual instances of the NSCharacterSet or NSMutableCharacterSet classes but of one of their private subclasses. Although a character set object's class is private, its interface is public, as declared by these abstract superclasses, NSCharacterSet and NSMutableCharacterSet. The character set classes adopt the NSCopying and NSMutableCopying protocols, making it convenient to convert a character set of one type to the other.

The NSCharacterSet class declares the programmatic interface for an object that manages a set of Unicode characters (see the NSString class cluster specification for information on Unicode). NSCharacterSet's principal primitive method, `characterIsMember:`, provides the basis for all other instance methods in its interface. A subclass of NSCharacterSet needs only to implement this method, plus `mutableCopyWithZone:`, for proper behavior. For optimal performance, a subclass should also override `bitmapRepresentation`, which otherwise works by invoking `characterIsMember:` for every possible Unicode value.

NSCharacterSet is "toll-free bridged" with its Cocoa Foundation counterpart, CFCharacterSet Reference. This means that the Core Foundation type is interchangeable in function or method calls with the bridged Foundation object. Therefore, in a method where you see an NSCharacterSet * parameter, you can pass a CFCharacterSetRef, and in a function where you see a CFCharacterSetRef parameter, you can pass an NSCharacterSet instance (you cast one type to the other to suppress compiler warnings). See Interchangeable Data Types for more information on toll-free bridging.

The mutable subclass of `NSCharacterSet` is `NSMutableCharacterSet`.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr5](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr7](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

- [7.6, page 8: News](#)

4.3.2 Methods

4.3.3 `alphanumericCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the categories Letters, Marks, and Numbers.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.alphanumericCharacterSet
MsgBox left(n,200) // show only first 200 chars
```

Notes: Informally, this set is the set of all characters used as basic units of alphabets, syllabaries, ideographs, and digits.

4.3.4 `bitmapRepresentation` as `MemoryBlock`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a memoryblock object encoding the receiver in binary format.

Notes: This format is suitable for saving to a file or otherwise transmitting or archiving.

A raw bitmap representation of a character set is a byte array of 2^{16} bits (that is, 8192 bytes). The value of the bit at position `n` represents the presence in the character set of the character with decimal Unicode value `n`.

4.3.5 `capitalizedLetterCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Titlecase Letters.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.capitalizedLetterCharacterSet
MsgBox n
```

4.3.6 characterIsMember(Character as Integer) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given character is in the receiver.

Example:

```
// get the Symbol font
dim n as NSFontMBS = NSFontMBS.fontWithName("Arial",10)

// what characters are defined for this font?
dim c as NSCharacterSetMBS = n.coveredCharacterSet

// is letter A part of this font?
MsgBox "A included: "+str(c.characterIsMember(asc("A")))
```

4.3.7 characterSetWithBitmapRepresentation(data as MemoryBlock) as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing characters determined by a given bitmap representation.

Notes: This method is useful for creating a character set object with data from a file or other external data source.

A raw bitmap representation of a character set is a byte array of 2^{16} bits (that is, 8192 bytes). The value of the bit at position *n* represents the presence in the character set of the character with decimal Unicode value *n*.

4.3.8 characterSetWithCharactersInString(aString as string) as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in a given string.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.characterSetWithCharactersInString("Hello World")
MsgBox n
```

Notes: A character set containing the characters in aString. Returns an empty character set if aString is empty.

4.3.9 characterSetWithContentsOfFile(aString as string) as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set read from the bitmap representation stored in the file a given path.

Notes: To read a bitmap representation from any file, use the NSData methoddataWithContentsOfFile and pass the result to characterSetWithBitmapRepresentation.

This method doesn't use filenames to check for the uniqueness of the character sets it creates. To prevent duplication of character sets in memory, cache them and make them available through an API that checks whether the requested set has already been loaded.

See also:

- 4.3.10 characterSetWithContentsOfFile(file as folderitem) as NSCharacterSetMBS 176

4.3.10 characterSetWithContentsOfFile(file as folderitem) as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set read from the bitmap representation stored in the file a given path.

Notes: To read a bitmap representation from any file, use the NSData.methoddataWithContentsOfFile and pass the result to characterSetWithBitmapRepresentation.

This method doesn't use filenames to check for the uniqueness of the character sets it creates. To prevent duplication of character sets in memory, cache them and make them available through an API that checks whether the requested set has already been loaded.

See also:

- 4.3.9 characterSetWithContentsOfFile(aString as string) as NSCharacterSetMBS 176

4.3.11 characterSetWithRange(r as NSRangeMBS) as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing characters with Unicode values in a given range.

Example:

```
dim r as NSRangeMBS = NSMakeRangeMBS(asc("a"),26) // all small letters
dim n as NSCharacterSetMBS = NSCharacterSetMBS.characterSetWithRange(r)
```

MsgBox n

Notes: r: A range of Unicode values.

r.location is the value of the first character to return; aRange.location + r.length-1 is the value of the last.

Returns a character set containing characters whose Unicode values are given by aRange. If aRange.length is 0, returns an empty character set.

This code excerpt creates a character set object containing the lowercase English alphabetic characters:

4.3.12 componentsSeparatedByCharactersInSet(s as string) as String()

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns an array containing substrings from the receiver that have been divided by characters in a given set.

Notes: self: A character set containing the characters to use to split the receiver. Must not be nil.
s: The text to process.

Returns an array of string containing substrings from the receiver that have been divided by characters in separator.

The substrings in the array appear in the order they did in the receiver. Adjacent occurrences of the separator characters produce empty strings in the result. Similarly, if the string begins or ends with separator characters, the first or last substring, respectively, is empty.

4.3.13 Constructor

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The constructor to create an empty character set.

4.3.14 controlCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the categories of Control or Format Char-

acters.

Notes: These characters are specifically the Unicode values U+0000 to U+001F and U+007F to U+009F.

4.3.15 `copy` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a copy of the character set.

4.3.16 `decimalDigitCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Decimal Numbers.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.decimalDigitCharacterSet  
MsgBox n
```

Notes: Informally, this set is the set of all characters used to represent the decimal values 0 through 9. These characters include, for example, the decimal digits of the Indic scripts and Arabic.

4.3.17 `decomposableCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing all individual Unicode characters that can also be represented as composed character sequences.

Notes: Returns a character set containing all individual Unicode characters that can also be represented as composed character sequences (such as for letters with accents), by the definition of "standard decomposition" in version 3.2 of the Unicode character encoding standard.

These characters include compatibility characters as well as pre-composed characters.

Note: This character set doesn't currently include the Hangul characters defined in version 2.0 of the Unicode standard.

4.3.18 hasMemberInPlane(thePlane as Integer) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver has at least one member in a given character plane.

Notes: This method makes it easier to find the plane containing the members of the current character set. The Basic Multilingual Plane is plane 0.

4.3.19 illegalCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing values in the category of Non-Characters or that have not yet been defined in version 3.2 of the Unicode standard.

4.3.20 invertedSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing only characters that don't exist in the receiver.

Notes: Inverting an immutable character set is much more efficient than inverting a mutable character set.

4.3.21 isSupersetOfSet(theOtherSet as NSCharacterSetMBS) as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver is a superset of another given character set.

Notes: Returns true if the receiver is a superset of theOtherSet, otherwise false.

Available in Mac OS X v10.2 and later.

4.3.22 letterCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the categories Letters and Marks.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.letterCharacterSet  
MsgBox n
```

Notes: Informally, this set is the set of all characters used as letters of alphabets and ideographs.

4.3.23 `longCharacterIsMember(theLongChar as Integer)` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether a given long character is a member of the receiver.

Notes: Returns true if theLongChar is in the receiver, otherwise false.

This method supports the specification of 32-bit characters.

Available in Mac OS X v10.2 and later.

4.3.24 `lowercaseLetterCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Lowercase Letters.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.lowercaseLetterCharacterSet
MsgBox n
```

Notes: Informally, this set is the set of all characters used as lowercase letters in alphabets that make case distinctions.

4.3.25 `mutableCopy` as `NSMutableCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a mutable copy of the character set.

4.3.26 `newlineCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the newline characters.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.newlineCharacterSet
MsgBox n
```

Notes: A character set containing the newline characters (U+000A–U+000D, U+0085). Available in Mac OS X v10.5 and later.

4.3.27 nonBaseCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Marks.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.nonBaseCharacterSet
MsgBox n
```

Notes: This set is also defined as all legal Unicode characters with a non-spacing priority greater than 0. Informally, this set is the set of all characters used as modifiers of base characters.

4.3.28 Operator_Convert as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a string containing all characters in this set.

Example:

```
dim r as NSRangeMBS = NSMakeRangeMBS(asc("a"),26) // all small letters
dim n as NSCharacterSetMBS = NSCharacterSetMBS.characterSetWithRange(r)
MsgBox n
```

Notes: This way you can use a character set directly with functions expecting a string like msgbox.

4.3.29 punctuationCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Punctuation.

Example:

`dim n as NSCharacterSetMBS = NSCharacterSetMBS.punctuationCharacterSet
MsgBox n`

Notes: Informally, this set is the set of all non-whitespace characters used to separate linguistic units in scripts, such as periods, dashes, parentheses, and so on.

4.3.30 `rangeOfCharacterFromSet(s as string, options as Integer = 0, searchRange as NSRangeMBS = nil) as NSRangeMBS`

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Finds and returns the range in the receiver of the first character from a given character set found in a given range with given options.

Notes: self: A character set. This value must not be nil.

s: The text to process:

options: A mask specifying search options. The following options may be specified by combining them with bitwise OR operator: `NSAnchoredSearch` (8), `NSBackwardsSearch` (4).

searchRange: Optional, the range in which to search. The range must not exceed the bounds of the receiver.

Raises a `NSEException` if search range is invalid.

Returns the range in the receiver of the first character found from set within search range. Returns a range of { `NSNotFound`, 0 } if none of the characters in aSet are found.

Because pre-composed characters in set can match composed character sequences in the receiver, the length of the returned range can be greater than 1. For example, if you search for "√" in the string "stru-@del", the returned range is { 3,2 }.

Special Considerations

This method detects all invalid ranges (including those with negative lengths). For applications linked against OS X v10.6 and later, this error causes an exception; for applications linked against earlier releases, this error causes a warning, which is displayed just once per application execution.

4.3.31 `stringByTrimmingCharactersInSet(s as string) as String`

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns a new string made by removing from both ends of the receiver characters contained in a given character set.

Notes: Returns a new string made by removing from both ends of the receiver characters contained in set.

If the receiver is composed entirely of characters from set, the empty string is returned.

Use `whitespaceCharacterSet` or `whitespaceAndNewlineCharacterSet` to remove whitespace around strings.

4.3.32 `symbolCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the category of Symbols.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.symbolCharacterSet
MsgBox n
```

Notes: These characters include, for example, the dollar sign (\$) and the plus (+) sign. Available in Mac OS X v10.3 and later.

4.3.33 `uppercaseLetterCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing the characters in the categories of Uppercase Letters and Titlecase Letters.

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.uppercaseLetterCharacterSet
MsgBox n
```

Notes: Informally, this set is the set of all characters used as uppercase letters in alphabets that make case distinctions.

4.3.34 `whitespaceAndNewlineCharacterSet` as `NSCharacterSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing only the whitespace characters space (U+0020) and tab (U+0009) and the newline and nextline characters (U+000A–U+000D, U+0085).

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.whitespaceAndNewlineCharacterSet
```

MsgBox n

4.3.35 whitespaceCharacterSet as NSCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a character set containing only the in-line whitespace characters space (U+0020) and tab (U+0009).

Example:

```
dim n as NSCharacterSetMBS = NSCharacterSetMBS.whitespaceCharacterSet  
MsgBox n
```

Notes: This set doesn't contain the newline or carriage return characters.

4.3.36 Properties

4.3.37 CharacterCount as Integer

Plugin Version: 24.1, Platform: macOS, Targets: All.

Function: Queries the character count for this character set.

Notes: (Read only property)

4.3.38 Handle as Integer

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.3.39 StringValue as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a string containing all characters in this set.

Example:

```
// get the Symbol font
```

```

dim n as NSFontMBS = NSFontMBS.fontWithName("Symbol",10)

// what characters are defined for this font?
dim c as NSCharacterSetMBS = n.coveredCharacterSet

// display a string with all the characters
MsgBox c.StringValue

```

Notes: Changed to property in v24.1.
(Read only property)

4.3.40 Constants

Constants

Constant	Value	Description
NSOpenStepUnicodeReservedBase	&hF400	A constant to specify lower bound for a Unicode character range reserved for Apple's corporate use. Specifies lower bound for a Unicode character range reserved for Apple's corporate use (the range is 0xF400–0xF8FF).

4.4 class NSCoderMBS

4.4.1 class NSCoderMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: The plugin class for NSCoder.

Example:

```
// make archiver
dim a as new NSKeyedArchiverMBS

// add a string
a.encodeString "Hello World", "Greeting"

// finish
a.finishEncoding

// query data
dim m as MemoryBlock = a.archiverData

// start unarchiver
dim u as new NSKeyedUnarchiverMBS(m)

// query and display a value
dim s as string = u.decodeString("Greeting")
MsgBox s
```

Notes: The NSCoder abstract class declares the interface used by concrete subclasses to transfer objects and other Objective-C data items between memory and some other format. This capability provides the basis for archiving (where objects and data items are stored on disk) and distribution (where objects and data items are copied between different processes or threads). The concrete subclasses provided by Foundation for these purposes are NSArchiver, NSUnarchiver, NSKeyedArchiver, NSKeyedUnarchiver, and NSPortCoder. Concrete subclasses of NSCoder are referred to in general as coder classes, and instances of these classes as coder objects (or simply coders). A coder object that can only encode values is referred to as an encoder object, and one that can only decode values as a decoder object.

NSCoder operates on objects, scalars, C arrays, structures, and strings, and on pointers to these types. It does not handle types whose implementation varies across platforms, such as union, void *, function pointers, and long chains of pointers. A coder object stores object type information along with the data, so an object decoded from a stream of bytes is normally of the same class as the object that was originally encoded into the stream. An object can change its class when encoded, however; this is described in Archives and Serializations Programming Guide.

For details of how to create a subclass of NSCoder, see "Subclassing NSCoder" in Archives and Serializations

Programming Guide.

The plugin implements this class because it is needed to encode and decode the window state. 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 13.2pr1](#)
- [MBS Real Studio Plugins, version 11.3pr2](#)
- [MBS Real Studio Plugins, version 11.2pr11](#)

4.4.2 Methods

4.4.3 `allowsKeyedCoding` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver supports keyed coding of objects.

Notes: The default implementation returns false. Concrete subclasses that support keyed coding, such as `NSKeyedArchiver`, need to override this method to return true.

4.4.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

4.4.5 `containsValueForKey(key as string)` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether an encoded value is available for a string.

4.4.6 `decodeBool(key as string)` as boolean

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes and returns a boolean value that was previously encoded with `encodeBool` and associated with the string key.

4.4.7 decodeBytes(key as string) as MemoryBlock

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes a buffer of data whose types are unspecified.

4.4.8 decodeCFOBJECTMBS(key as string) as Variant

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

Function: Decodes CFOBJECT object.

Notes: This is a special convenience methods to decode CF* classes which are bridged to Cocoa objects. Like CFDictionary, CFURL, CFData, CFString, CFNumber, CFArray and a few others.

4.4.9 decodeDictionary(key as string) as Dictionary

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Decodes a dictionary.

4.4.10 decodeDouble(key as string) as Double

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes and returns a double value that was previously encoded with either encodeFloat or encodeDouble and associated with the string key.

4.4.11 decodeFloat(key as string) as single

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes and returns a float value that was previously encoded with encodeFloat or encodeDouble and associated with the string key.

4.4.12 decodeInt32(key as string) as Int32

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes and returns a 32-bit integer value that was previously encoded with encodeInt, encodeInteger, encodeInt32, or encodeInt64 and associated with the string key.

Notes: If the encoded integer does not fit into a 32-bit integer, the method raises an NSRangeException.

4.4.13 decodeInt64(key as string) as Int64

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Decodes and returns a 64-bit integer value that was previously encoded with encodeInt, encodeInteger, encodeInt32, or encodeInt64 and associated with the string key.

4.4.14 decodeNSURLFile(key as string) as folderitem

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

Function: Decodes an NSURL and returns folderitem.

4.4.15 decodeNSURLString(key as string) as String

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

Function: Decodes an NSURL and returns string.

4.4.16 decodePoint(key as string) as NSPointMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Decodes a point object.

4.4.17 decodeRect(key as string) as NSRectMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Decodes a rectangle.

4.4.18 decodeSize(key as string) as NSSizeMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Decodes a size object.

4.4.19 decodeString(key as string) as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Decodes a string.

4.4.20 decodeTypedStream(Data as MemoryBlock) as variant

Plugin Version: 23.2, Platform: macOS, Targets: All.

Function: Decodes an old and deprecated typedstream data block.

Example:

```
Dim f As FolderItem = SpecialFolder.Desktop.Child("test.bin")
Dim b As BinaryStream = BinaryStream.Open(f)
Dim s As String = b.Read(B.Length)
Dim v As Variant = NSCoderMBS.decodeTypedStream(s)
```

Break // check in debugger

4.4.21 encodeBool(value as boolean, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes value and associates it with the string key.

4.4.22 encodeBytes(value as MemoryBlock, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes a buffer of data whose types are unspecified.

4.4.23 encodeCFObjectMBS(value as Variant, key as string)

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

Function: Encodes a CFObject.

Notes: This is a special convenience methods to encode CF* classes which are bridged to Cocoa objects.

Like CFDictionary, CFURL, CFData, CFString, CFNumber, CFArray and a few others.

4.4.24 encodeDictionary(value as Dictionary, key as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Encodes a dictionary.

Notes: Not all objects can be inside the dictionary.

Cocoa objects, string, booleans and numbers are okay.

4.4.25 encodeDouble(value as Double, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes value and associates it with the string key.

4.4.26 encodeFloat(value as single, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes value and associates it with the string key.

4.4.27 encodeInt32(value as Int32, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes the 32-bit integer value and associates it with the string key.

4.4.28 encodeInt64(value as Int64, key as string)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Encodes the 64-bit integer value and associates it with the string key.

4.4.29 encodeNSURLFile(value as folderitem, key as string)

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

Function: Encodes a folderitem as NSURL.

4.4.30 encodeNSString(value as String, key as string)

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

Function: Encodes a string with URL as NSURL object.

4.4.31 encodePoint(value as NSPointMBS, key as string)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Encodes a point object.

4.4.32 encodeRect(value as NSRectMBS, key as string)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Encodes a rectangle.

4.4.33 encodeSize(value as NSSizeMBS, key as string)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Encodes a size object.

4.4.34 encodeString(value as string, key as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Encodes a string.

4.4.35 systemVersion as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: During encoding, this method should return the system version currently in effect.

Notes: During decoding, this method should return the version that was in effect when the data was encoded.

By default, this method returns the current system version, which is appropriate for encoding but not for decoding. Subclasses that implement decoding must override this method to return the system version of the data being decoded.

4.4.36 Properties

4.4.37 Handle as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: The internal reference for this object.

Notes: (Read and Write property)

4.5 class NSCursorMBS

4.5.1 class NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Instances of the NSCursor class manage the appearance of the cursor.

Example:

```
Dim im As NSImageMBS
Dim p As Picture
Dim m As Picture
Dim theCursor As NSCursorMBS
Dim test As Boolean

// create a blue ball picture
p = New Picture(16,16,32)
p.Graphics.ForeColor = &c0000FF
p.Graphics.FillRect 0,0,16,16

m = New Picture(16,16,32)
m.Graphics.ForeColor = &c000000
m.Graphics.Filloval 0,0,16,16

// Create a new NSImage
im=New NSImageMBS(p,m)

// Create a cursor from the NSImage
theCursor=New NSCursorMBS(im, 10, 10)

Title = Str(theCursor.Handle)

// Make this the active cursor
theCursor.set

// display picture
p.Mask.Graphics.DrawPicture m,0,0
Backdrop = p

// so you see it for a second before RB resets the cursor
DelayMBS 1.0
```

Notes: In Cocoa, you can change the currently displayed cursor based on the position of the mouse over one of your views. You might use this technique to provide visual feedback about what actions the user can take with the mouse. For example, you might display one of the resize cursors whenever the mouse moves over a portion of your view that acts as a custom resizing handle. To set this up, you associate a cursor

object with one or more cursor rectangles in the view.

Cursor rectangles are a specialized type of tracking rectangles, which are used to monitor the mouse location in a view. Views implement cursor rectangles using tracking rectangles but provide methods for setting and refreshing cursor rectangles that are distinct from the generic tracking rectangle interface. For information on how to set up cursor rectangles, see "Handling Tracking-Rectangle and Cursor-Update Events in Views".
Blog Entries

- [MBS Xojo Plugins, version 20.1pr3](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)
- [MBS REALbasic plug-in 9.6](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.6](#)

4.5.2 Methods

4.5.3 arrowCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the default cursor, the arrow cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.arrowCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The default cursor, a slanted arrow with its hot spot at the tip. The arrow cursor is the one you're used to seeing over buttons, scrollers, and many other objects in the window system.

4.5.4 closedHandCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the closed-hand system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.closedHandCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

4.5.5 Constructor(image as NSImageMBS, foregroundColorHint as NSColorMBS, backgroundColorHint as NSColorMBS, HotSpotX as Double, HotSpotY as Double)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Initializes the cursor with the specified image and hot spot.

See also:

- 4.5.6 Constructor(image as NSImageMBS, HotSpotX as Double, HotSpotY as Double) 196

4.5.6 Constructor(image as NSImageMBS, HotSpotX as Double, HotSpotY as Double)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Initializes the cursor with the specified image and hot spot.

See also:

- 4.5.5 Constructor(image as NSImageMBS, foregroundColorHint as NSColorMBS, backgroundColorHint as NSColorMBS, HotSpotX as Double, HotSpotY as Double) 196

4.5.7 contextualMenuCursor as NSCursorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the contextual menu system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.contextualMenuCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Available in Mac OS X v10.6 and later.

4.5.8 crosshairCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the cross-hair system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.crosshairCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: This cursor is used for situations when precise location is required (where the lines cross is the hot spot).

4.5.9 currentCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the application's current cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.currentCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The top cursor on the application's cursor stack. This cursor may not be the visible cursor on the screen if a different application is currently active.

4.5.10 currentSystemCursor as NSCursorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the current system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.currentSystemCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Returns a cursor whose image and hot spot match those of the currently-displayed cursor on the system.

This method returns the current system cursor regardless of which application set the cursor, and whether

Cocoa or Carbon APIs were used to set it.
This method replaces the now deprecated QDGetCursorData function.
Available in Mac OS X v10.6 and later.

4.5.11 disappearingItemCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a cursor indicating that the current operation will result in a disappearing item.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.disappearingItemCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The system cursor that indicates that the current operation will result in a disappearing item (for example, when dragging an item from the dock or a toolbar).

Available in Mac OS X v10.3 and later.

4.5.12 dragCopyCursor as NSCursorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a cursor indicating that the current operation will result in a copy action.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.dragCopyCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Available in Mac OS X v10.6 and later.

4.5.13 dragLinkCursor as NSCursorMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a cursor indicating that the current operation will result in a link action.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.dragLinkCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Available in Mac OS X v10.6 and later.

4.5.14 hide

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Makes the current cursor invisible.

Notes: If another cursor becomes current, that cursor will be invisible, too. It will remain invisible until you invoke the unhide method.

hide overrides setHiddenUntilMouseMoves.

4.5.15 hotSpotX as Double

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the position of the cursor's hot spot.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.arrowCursor

MsgBox str(c.hotSpotX)+"/" +str(c.hotSpotY)
```

Notes: The point describing the position of the hot spot, specified according to the cursor's flipped coordinate system.

For a more complete explanation, see the class description.

Note that an NSCursor object is immutable: you cannot change its hot spot after it's created. Instead, use the Constructor to create a new cursor with the new settings.

4.5.16 hotSpotY as Double

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the position of the cursor's hot spot.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.arrowCursor

MsgBox str(c.hotSpotX)+"/" +str(c.hotSpotY)
```

Notes: The point describing the position of the hot spot, specified according to the cursor's flipped coordinate system.

For a more complete explanation, see the class description.

Note that an NSCursor object is immutable: you cannot change its hot spot after it's created. Instead, use the Constructor to create a new cursor with the new settings.

4.5.17 IBeamCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a cursor that looks like a capital I with a tiny crossbeam at its middle.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.IBeamCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The I-beam cursor. This is the cursor that you're used to seeing over editable or selectable text. The I-beam cursor's default hot spot is where the crossbeam intersects the I.

4.5.18 IBeamCursorForVerticalLayout as NSCursorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns a cursor that looks like a capital I with a tiny crossbeam at its middle.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.IBeamCursorForVerticalLayout
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Available in Mac OS X 10.7.

4.5.19 image as NSImageMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's image.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.arrowCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The cursor image or nil if none exists

Note that an NSCursor object is immutable: you cannot change its image after it's created. Instead, use the constructor to create a new cursor with the new settings.

4.5.20 isSetOnMouseEntered as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether the receiver becomes current on receiving a mouseEntered message.

Notes: True if the receiver will become current when it receives a mouseEntered message; otherwise, false.

To receive such a message, the receiver must first be assigned a cursor rectangle. This assignment can be made using the NSView method addCursorRect. For a more complete explanation, see the class description.

4.5.21 isSetOnMouseExited as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether the receiver becomes current when it receives a `mouseExited:` message.

Notes: True if the receiver becomes current when it receives a `mouseExited:` message; otherwise, false.

To receive such a message, the receiver must first be assigned a cursor rectangle. This assignment can be made using the `NSView` method `addCursorRect`. For a more complete explanation, see the class description.

4.5.22 `mouseEntered(e as NSEventMBS)`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Automatically sent to the receiver when the cursor enters a cursor rectangle owned by the receiver.

Notes: If used after `setOnMouseEntered` has been called with an argument of true, `mouseEntered` can make the receiver the current cursor.

In your programs, you won't invoke `mouseEntered` explicitly. It's only included in the class interface so you can override it.

For a more complete explanation, see "Handling Tracking-Rectangle and Cursor-Update Events in Views" and the `NSView` method `addTrackingRect`

4.5.23 `mouseExited(e as NSEventMBS)`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Automatically sent to the receiver when the cursor exits a cursor rectangle owned by the receiver.

Notes: Like `mouseEntered`, this message is part of the class interface only so you can override it.

For a more complete explanation, see "Handling Tracking-Rectangle and Cursor-Update Events in Views" and the `NSView` method `addTrackingRect`.

4.5.24 `openHandCursor as NSCursorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the open-hand system cursor.

Example:

```
// shows cursor in window  
dim c as NSCursorMBS = NSCursorMBS.openHandCursor
```

```
dim i as NSImageMBS = c.image  
window1.Backdrop=i.CopyPictureWithMask
```

Notes: Available in Mac OS X v10.3 and later.

4.5.25 operationNotAllowedCursor as NSCursorMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the operation not allowed cursor.

Notes: This cursor indicates that the operation that is being attempted, perhaps dragging to an item that can't accept the drag type, is being denied.

Available in Mac OS X v10.6 and later.

4.5.26 pointingHandCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the pointing-hand system cursor.

Example:

```
// shows cursor in window  
dim c as NSCursorMBS = NSCursorMBS.pointingHandCursor  
dim i as NSImageMBS = c.image  
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The pointing-hand cursor. The tip of the pointing finger is the hot spot.

4.5.27 pop

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sends a pop message to the receiver's class.

See also:

- 4.5.28 pop

4.5.28 pop

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Pops the current cursor off the top of the stack.

Notes: The new object on the top of the stack becomes the current cursor. If the current cursor is the only cursor on the stack, this method does nothing.

See also:

- 4.5.27 pop

203

4.5.29 push

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Puts the receiver on top of the cursor stack and makes it the current cursor.

4.5.30 resizeModeCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-down system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.resizeDownCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-down cursor. This cursor is used when moving or resizing an object to indicate that the user can move only in the indicated direction.

Available in Mac OS X v10.3 and later.

4.5.31 resizeModeCursor as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-left system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.resizeLeftCursor
```

```
dim i as NSImageMBS = c.image  
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-left cursor. This cursor is used when moving or resizing an object to indicate that the user can move only in the indicated direction.

Available in Mac OS X v10.3 and later.

4.5.32 `resizeLeftRightCursor` as `NSCursorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-left-and-right system cursor.

Example:

```
// shows cursor in window  
dim c as NSCursorMBS = NSCursorMBS.resizeLeftRightCursor  
dim i as NSImageMBS = c.image  
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-left-and-right cursor. This cursor is used when moving or resizing an object and the object can be moved left or right.

Available in Mac OS X v10.3 and later.

4.5.33 `resizeRightCursor` as `NSCursorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-right system cursor.

Example:

```
// shows cursor in window  
dim c as NSCursorMBS = NSCursorMBS.resizeRightCursor  
dim i as NSImageMBS = c.image  
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-right cursor. This cursor is used when moving or resizing an object to indicate that the user can move only in the indicated direction.

Available in Mac OS X v10.3 and later.

4.5.34 `resizeUpCursor` as `NSCursorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-up system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.resizeUpCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-up cursor. This cursor is used when moving or resizing an object to indicate that the user can move only in the indicated direction.

Available in Mac OS X v10.3 and later.

4.5.35 `resizeUpDownCursor` as `NSCursorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the resize-up-and-down system cursor.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.resizeUpDownCursor
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

Notes: The resize-up-and-down cursor. This cursor is used when moving or resizing an object and the object can be moved up or down.

Available in Mac OS X v10.3 and later.

4.5.36 ringCursorWithDiameter(diameter as Double) as NSCursorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Creates a ring cursor with the given size.

Example:

```
// shows cursor in window
dim c as NSCursorMBS = NSCursorMBS.ringCursorWithDiameter(20)
dim i as NSImageMBS = c.image
window1.Backdrop=i.CopyPictureWithMask
```

4.5.37 set

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Makes the receiver the current cursor.

Notes: If your application is not the front application, the system will ignore this set message!

4.5.38 setHiddenUntilMouseMoves(value as boolean)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets whether the cursor is hidden until the mouse moves.

Notes: value: True to hide the cursor until one of the following occurs:
The mouse moves.

You invoke the method again, with flag set to false.

Do not try to counter this method by invoking unhide. The results are undefined.

4.5.39 setOnMouseEntered(flag as boolean)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: specifies whether the receiver accepts mouseEntered: events.

Notes: True if the receiver accepts future mouseEntered event messages; otherwise it ignores them.

Accepting mouseEntered event messages allows the cursor to be made the current cursor when the cursor enters a view's cursor rectangle.

4.5.40 `setOnMouseExited(flag as boolean)`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Sets whether the receiver accepts mouseExited events.

Notes: flag: True if the receiver accepts future mouseExited: event messages; otherwise it ignores them.

Accepting mouseExited event messages allows the cursor to be made the current cursor when the cursor exits a view's cursor rectangle.

4.5.41 `unhide`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Negates an earlier call to hide by showing the current cursor.

4.5.42 `Properties`

4.5.43 `Handle as Integer`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: The internal used NSCursor reference.

Notes: (Read and Write property)

4.6 class NSDateComponentsMBS

4.6.1 class NSDateComponentsMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: NSDateComponents encapsulates the components of a date in an extendable, object-oriented manner.

Notes: It is used to specify a date by providing the temporal components that make up a date and time: hour, minutes, seconds, day, month, year, and so on. It can also be used to specify a duration of time, for example, 5 hours and 16 minutes. An NSDateComponents object is not required to define all the component fields. When a new instance of NSDateComponents is created the date components are set to NSUndefinedDateComponent.

Blog Entries

- [News from the MBS Xojo Plugins Version 20.1](#)
- [MBS Xojo Plugins, version 20.1pr1](#)
- [MBS Real Studio Plugins, version 12.3pr11](#)

4.6.2 Methods

4.6.3 Constructor

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Creates new date components object.

4.6.4 copy as NSDateComponentsMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns a copy of the date component object.

4.6.5 isValidDateInCalendar(calendar as NSCalendarMBS) as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the current combination of properties represents a date which exists in the specified calendar.

Notes: calendar: The calendar for which to use in the calculation.

Returns yes if the date corresponding to the receiver's values is valid and exists in the given calendar, otherwise no.

If the `timeZone` property is set on the receiver, the time zone property value is used.

This property should not be used for `NSDateComponents` objects that represent relative quantities of calendar components.

4.6.6 `NSUndefinedDateComponent` as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: This constant specifies that an `NSDateComponents` component is undefined.

4.6.7 `Print`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Writes the date components to debug output.

Notes: This may help for debugging and you see output in console app.

4.6.8 `Properties`

4.6.9 `calendar` as `NSCalendarMBS`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The calendar.

Notes: (Read and Write property)

4.6.10 `date` as `date`

Plugin Version: 12.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The date.

Notes: Available in Mac OS X v10.7 and later.

(Read only property)

4.6.11 `dateTime` as `Date`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The date.

Notes: Available in Mac OS X v10.7 and later.
(Read only property)

4.6.12 `day` as `Integer`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of day units.

Notes: (Read and Write property)

4.6.13 `description` as `string`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns description of the calendar object.

Notes: (Read only property)

4.6.14 `era` as `Integer`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of era units.

Notes: (Read and Write property)

4.6.15 `Handle` as `Integer`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.6.16 `hour` as `Integer`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of hour units.

Notes: (Read and Write property)

4.6.17 isLeapMonth as Boolean

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Whether it is leap month.

Notes: Available on Mac OS X 10.8.

(Read and Write property)

4.6.18 isValidDate as Boolean

Plugin Version: 20.1, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the current combination of properties represents a date which exists in the current calendar.

Notes: If the timeZone property is set on the receiver, the time zone property value is used. If the calendar property is not set on the receiver, nil is returned.

(Read only property)

4.6.19 minute as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of minute units.

Notes: (Read and Write property)

4.6.20 month as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of month units.

Notes: (Read and Write property)

4.6.21 quarter as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of quarters.

Notes: (Read and Write property)

4.6.22 second as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of second units.

Notes: (Read and Write property)

4.6.23 timeZone as NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The time zone.

Notes: (Read and Write property)

4.6.24 week as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of week units.

Notes: (Read and Write property)

4.6.25 weekday as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of weekday units.

Notes: Weekday units are the numbers 1 through n, where n is the number of days in the week. For example, in the Gregorian calendar, n is 7 and Sunday is represented by 1.

(Read and Write property)

4.6.26 weekdayOrdinal as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The ordinal number of weekday units.

Notes: Weekday ordinal units represent the position of the weekday within the next larger calendar unit, such as the month. For example, 2 is the weekday ordinal unit for the second Friday of the month.

(Read and Write property)

4.6.27 weekOfMonth as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The week of the month.

Notes: (Read and Write property)

4.6.28 weekOfYear as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The week of the year.

Notes: (Read and Write property)

4.6.29 year as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The number of year unit.

Notes: (Read and Write property)

4.6.30 yearForWeekOfYear as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The year for the week of the year.

Notes: (Read and Write property)

4.7 class NSDirectoryEnumeratorMBS

4.7.1 class NSDirectoryEnumeratorMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: An NSDirectoryEnumerator object enumerates the contents of a directory, returning the pathnames of all files and directories contained within that directory.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
dim f as string = d.nextObject

while len(f)>0
List.AddRow f

f=d.nextObject
wend
```

Notes: These pathnames are relative to the directory.

An enumeration is recursive, including the files of all subdirectories, and crosses device boundaries. An enumeration does not resolve symbolic links, or attempt to traverse symbolic links that point to directories. Subclass of the NSEnumeratorMBS class.

Blog Entries

- [NSBundleMBS and NSDirectoryEnumeratorMBS](#)

4.7.2 Methods

4.7.3 Constructor(folder as folderitem)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an NSDirectoryEnumerator object that enumerates the contents of the directory at a given path.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)

MsgBox d.nextObject // shows ".DS_Store" or some other file name
```

Notes: An NSDirectoryEnumerator object that enumerates the contents of the directory at path. If path is a filename, the method returns an enumerator object that enumerates no files—the first call to

nextObject will return nil.

Because the enumeration is deep—that is, it lists the contents of all subdirectories—this enumerator object is useful for performing actions that involve large file-system subtrees. This method does not resolve symbolic links encountered in the traversal process, nor does it recurse through them if they point to a directory.

See also:

- 4.7.4 Constructor(path as string) 216

4.7.4 Constructor(path as string)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an NSDirectoryEnumerator object that enumerates the contents of the directory at a given path.

Example:

```
dim d as new NSDirectoryEnumeratorMBS("/Applications")
dim f as FolderItem = d.nextFile
```

```
while f<>Nil
List.AddRow f.Name
```

```
// if this is a folder, we skip the sub folders
d.skipDescendents
```

```
f=d.nextFile
wend
```

Notes: An NSDirectoryEnumerator object that enumerates the contents of the directory at path. If path is a filename, the method returns an enumerator object that enumerates no files—the first call to nextObject will return nil.

Because the enumeration is deep—that is, it lists the contents of all subdirectories—this enumerator object is useful for performing actions that involve large file-system subtrees. This method does not resolve symbolic links encountered in the traversal process, nor does it recurse through them if they point to a directory.

See also:

- 4.7.3 Constructor(folder as folderitem) 215

4.7.5 Destructor

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The destructor for this class.

4.7.6 directoryAttributes as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an Dictionary object that contains the attributes of the directory at which enumeration started.

4.7.7 fileAttributes as dictionary

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an object that contains the attributes of the most recently returned file or subdirectory (as referenced by the pathname).

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
dim f as FolderItem = d.nextFile
```

```
while f<>Nil
List.AddRow f.Name
```

```
dim di as Dictionary = d.fileAttributes
dim size as int64 = di.Value(d.NSFileSize)
```

```
List.AddRow f.Name+" (" +str(size)+" Bytes)"
```

```
f=d.nextFile
wend
```

4.7.8 level as Integer

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the number of levels deep the current object is in the directory hierarchy being enumerated.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
dim f as FolderItem = d.nextFile
```

```
while f<>Nil
```

```
List.AddRow f.Name+" "+str(d.level)
```

```
f=d.nextFile
wend
```

Notes: Available in Mac OS X v10.6 and later.

4.7.9 nextFile as folderitem

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the next folderitem from the collection being enumerated.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
dim f as string = d.nextObject
```

```
while len(f)>0
List.AddRow f
```

```
f=d.nextObject
wend
```

Notes: The next folderitem from the collection being enumerated, or nil when all objects have been enumerated.

Same as nextObject, but returns a folderitem.

4.7.10 NSFileAppendOnly as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates whether the file is read-only.

4.7.11 NSFileBusy as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates whether the file is busy.

4.7.12 NSFileCreationDate as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's creation date.

4.7.13 NSFileDeviceIdentifier as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the identifier for the device on which the file resides.

4.7.14 NSFileExtensionHidden as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates whether the file's extension is hidden.

4.7.15 NSFileGroupOwnerAccountID as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's group ID.

4.7.16 NSFileGroupOwnerAccountName as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the group name of the file's owner.

4.7.17 `NSFileHFSCreatorCode` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's HFS creator code.

4.7.18 `NSFileHFSTypeCode` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's HFS type code.

4.7.19 `NSFileImmutable` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates whether the file is mutable.

4.7.20 `NSFileModificationDate` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's last modified date.

4.7.21 `NSFileOwnerAccountID` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's owner's account ID.

4.7.22 `NSFileOwnerAccountName` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the name of the file's owner.

4.7.23 NSFilePosixPermissions as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's Posix permissions.

4.7.24 NSFileReferenceCount as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's reference count.

4.7.25 NSFileSize as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
dim f as FolderItem = d.nextFile

while f<>Nil
List.AddRow f.Name

dim di as Dictionary = d.fileAttributes
dim size as int64 = di.Value(d.NSFileSize)

List.AddRow f.Name+" (" +str(size)+" Bytes)"

f=d.nextFile
wend
```

Notes: The key in a file attribute dictionary whose value indicates the file's size in bytes.

4.7.26 `NSFileSystemFileNumber` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's filesystem file number.

4.7.27 `NSFileSystemFreeNodes` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file system attribute dictionary whose value indicates the number of free nodes in the file system.

4.7.28 `NSFileSystemFreeSize` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file system attribute dictionary whose value indicates the amount of free space on the file system.

4.7.29 `NSFileSystemNodes` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file system attribute dictionary whose value indicates the number of free nodes in the file system.

4.7.30 `NSFileSystemNumber` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file system attribute dictionary whose value indicates the filesystem number of the file system.

4.7.31 *NSFileSystemSize* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file system attribute dictionary whose value indicates the size of the file system.

4.7.32 *NSFileType* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the attributes dictionaries.

Notes: The key in a file attribute dictionary whose value indicates the file's type.

4.7.33 *NSFileTypeBlockSpecial* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Block special file

4.7.34 *NSFileTypeCharacterSpecial* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Character special file

4.7.35 *NSFileTypeDirectory* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Directory

4.7.36 *NSFileTypeRegular* as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Regular file

4.7.37 `NSFileTypeSocket` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Socket

4.7.38 `NSFileTypeSymbolicLink` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Symbolic link

4.7.39 `NSFileTypeUnknown` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: One of the constants for the file type property.

Notes: Unknown

4.7.40 `Path` as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The path used in the constructor.

4.7.41 `skipDescendents`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Causes the receiver to skip recursion into the most recently obtained subdirectory.

4.8 class NSEnumeratorMBS

4.8.1 class NSEnumeratorMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: NSEnumerator is an abstract class, instances of whose subclasses enumerate collections of other objects, such as arrays and dictionaries.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
MsgBox d.nextObject
```

Notes: You send nextObject repeatedly to a newly created NSEnumerator object to have it return the next object in the original collection. When the collection is exhausted, nil is returned. You cannot "reset" an enumerator after it has exhausted its collection. To enumerate a collection again, you need a new enumerator. This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [NSBundleMBS and NSDirectoryEnumeratorMBS](#)

4.8.2 Methods

4.8.3 allObjects as Variant()

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns an array of objects the receiver has yet to enumerate.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
```

```
dim a(-1) as Variant = d.allObjects
dim lines(-1) as string
```

```
for each v as Variant in a
lines.Append v
next
```

```
MsgBox Join(lines, EndOfLine) // shows all stuff on desktop
```

Notes: Put another way, the array returned by this method does not contain objects that have already been enumerated with previous nextObject messages.

Invoking this method exhausts the enumerator's collection so that subsequent invocations of nextObject

return nil.

4.8.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

4.8.5 nextObject as Variant

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the next object from the collection being enumerated.

Example:

```
dim d as new NSDirectoryEnumeratorMBS(SpecialFolder.Desktop)
```

```
MsgBox d.nextObject // shows ".DS_Store" or some other file name
```

Notes: The next object from the collection being enumerated, or nil when all objects have been enumerated.

4.8.6 Properties

4.8.7 Handle as Integer

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The internal reference to the NSEnumerator object.

Notes: (Read and Write property)

4.9 class NSEventMBS

4.9.1 class NSEventMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: The Cocoa class for an user event.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [MBS Xojo Plugins, version 24.1pr3](#)
- [MBS Xojo Plugins, version 23.5pr7](#)
- [News from the MBS Xojo Plugins Version 22.3](#)
- [MBS Xojo Plugins, version 22.3pr2](#)
- [MBS Xojo Plugins, version 18.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr5](#)
- [MBS Real Studio Plugins, version 13.1pr16](#)
- [MBS Real Studio Plugins, version 11.3pr8](#)
- [Gestures on Mac OS X](#)

4.9.2 Methods

4.9.3 allTouches as NSTouchMBS()

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns all touch objects associated with the event.

Example:

```
Dim e As NSEventMBS = NSApplicationMBS.sharedApplication.currentEvent
Dim touches() As NSTouchMBS = e.allTouches
```

Break

Notes: If the touches originate in different views or windows, each NSTouchMBS object may have a different responder object.

4.9.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

4.9.5 doubleClickInterval as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the time, in seconds, in which a second mouse click must occur in order to be considered a double click.

Notes: This is a system setting, overriding this method will have no effect.
Available in Mac OS X v10.6 and later.

4.9.6 eventWithCGEvent(CGEventRef as Integer) as NSEventMBS

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

Function: Creates and returns an event object that is based on a Core Graphics type of event.

Notes: The returned object retains the CGEventRef object (cgEvent) until it (the Xojo and Objective-C object) is freed—it then releases the CGEventRef object. If no Cocoa event corresponds to the CGEventRef object, this method returns nil.

4.9.7 isMouseCoalescingEnabled as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Indicates whether mouse-movement event coalescing is enabled.

Notes: Available in Mac OS X v10.5 and later.

4.9.8 keyEvent(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, characters as String, charactersIgnoringModifiers as String, isARepet as boolean, keyCode as Integer) as NSEventMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Returns a new NSEventMBS object describing a key event.

Notes: type: One of the following event-type constants: `NSKeyDown`, `NSKeyUp`, `NSFlagsChanged`. If anything else is specified, an `NSInternalInconsistencyException` is raised.

location: The cursor location in the base coordinate system of the window specified by windowNum.

modifierFlags: An integer bit field containing any of the modifier key masks.

timeStamp: The time the event occurred in seconds since system startup.

windowNumber: An integer that identifies the window device associated with the event, which is associated with the NSWindow that will receive the event.

characters: A string of characters associated with the key event. Though most key events contain only one character, it is possible for a single keypress to generate a series of characters.

unmodCharacters: The string of characters generated by the key event as if no modifier key had been pressed (except for Shift). This argument is useful for getting the „Áúbasic,Àù key value in a hardware-independent manner.

isARepeat: True if the key event is a repeat caused by the user holding the key down, false if the key event is new.

keyCode: A number that identifies the keyboard key associated with the key event. Its value is hardware-independent.

Returns the created NSEvent instance or nil if the instance could not be created.

4.9.9 keyRepeatDelay as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the length of time a key must be held down in order to generate the first key repeat event.

Notes: This is a system setting, overriding this method will have no effect.
Available in Mac OS X v10.6 and later.

4.9.10 keyRepeatInterval as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the length between subsequent key repeat events being posted.

Notes: This is a system setting, overriding this method will have no effect.
Available in Mac OS X v10.6 and later.

4.9.11 modifierFlagsGlobal as UInt32

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the currently pressed modifier flags.

Notes: This returns the state of devices combined with synthesized events at the moment, independent of which events have been delivered via the event stream.
Available in Mac OS X v10.6 and later.

4.9.12 `mouseEvent`(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, eventNumber as Integer, clickCount as Integer, pressure as Single) as NSEventMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Returns a new NSEvent object describing a mouse-down, -up, -moved, or -dragged event.

Notes: type: One of the modifier key masks described in NSEventType, or an NSInternalInconsistencyException is raised.

location: The cursor location in the base coordinate system of the window specified by windowNumber.

flags: An integer bit field containing any of the modifier key masks described in Getting Unicode Values, combined using the C bitwise OR operator.

timeStamp: The time the event occurred in seconds since system startup.

windowNumber: An integer that identifies the window device associated with the event, which is associated with the NSWindow that will receive the event.

eventNumber: An identifier for the new event. It,Äôs normally taken from a counter for mouse events, which continually increases as the application runs.

clickNumber: The number of mouse clicks associated with the mouse event.

pressure: A value from 0.0 to 1.0 indicating the pressure applied to the input device on a mouse event, used for an appropriate device such as a graphics tablet. For devices that aren,Äôt pressure-sensitive, the value should be either 0.0 or 1.0.

Returns the created NSEventMBS instance or nil if the instance could not be created.

4.9.13 `mouseLocation` as NSPointMBS

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Reports the current mouse position in screen coordinates.

Notes: This method is similar to the NSWindow method `mouseLocationOutsideOfEventStream`. It returns the location regardless of the current event or pending events. The difference between these methods is that `mouseLocationOutsideOfEventStream` returns a point in the receiving window's coordinates and `mouseLocation` returns the same information in screen coordinates.

Note: The y coordinate in the returned point starts from a base of 1, not 0.

4.9.14 otherEvent(Type as Integer, LocationX as Double, LocationY as Double, modifierFlags as Integer, timeStamp as double, windowNumber as Integer, SubType as Integer, Data1 as Integer, Data2 as Integer) as NSEventMBS

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Returns a new NSEventMBS object describing a custom event.

Notes: type: One of the following event-type constants: NSAppKitDefined, NSSystemDefined, NSApplicationDefined or NSPeriodic.

If type is anything else, an NSInternalInconsistencyException is raised. Your code should only create events of type NSApplicationDefined.

location: The cursor location in the base coordinate system of the window specified by windowNum.

flags: An integer bit field containing any of the modifier key masks.

time: The time the event occurred in seconds since system startup.

windowNumber: An integer that identifies the window device associated with the event, which is associated with the NSWindow that will receive the event.

subtype: A numeric identifier that further differentiates custom events of types NSAppKitDefined, NSSystemDefined, and NSApplicationDefined. NSPeriodic events don't use this attribute.

data1: Additional data associated with the event. NSPeriodic events don't use these attributes.

data2: Additional data associated with the event. NSPeriodic events don't use these attributes.

Returns the created NSEvent object or nil if the object couldn't be created.

4.9.15 pressedMouseButtons as UInt32

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the indices of the currently depressed mouse buttons.

Notes: The indices of the currently depressed mouse buttons.

A return value of 1 «0 corresponds to left the mouse, 1 «1 corresponds to the right mouse, 1«n, n >=2 to other mouse buttons.

This returns the state of devices combined with synthesized events at the moment, independent of which events have been delivered via the event stream, so this method is not suitable for tracking.

Available in Mac OS X v10.6 and later.

4.9.16 setMouseCoalescingEnabled(Value as boolean)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether mouse-movement event coalescing is enabled.

Notes: This method affects mouse-moved, mouse-dragged, and tablet events. Mouse-movement event coalescing is enabled by default.

Available in Mac OS X v10.5 and later.

4.9.17 touchesForView(view as NSViewMBS) as NSTouchMBS()

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns the touch objects from the event that belong to the specified view.

4.9.18 touchesForWindow(win as NSWindowMBS) as NSTouchMBS()

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: iOS only.

Function: Returns the touch objects from the event that belong to the specified window.

4.9.19 Properties

4.9.20 absoluteX as Integer

Plugin Version: 8.7, Platform: macOS, Targets: Desktop only.

Function: Reports the absolute x coordinate of a pointing device on its tablet at full tablet resolution.

Notes: For the coordinate to be valid, the receiver should represent an event generated by a tablet pointing device (otherwise 0 is returned). This method is valid only for mouse events with a subtype of NSTabletPointEventSubtype and for events of type NSTabletPoint. Use this value if you want to scale from tablet location to screen location yourself; otherwise use the class method mouseLocation or the instance method locationInWindow.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.21 absoluteY as Integer

Plugin Version: 8.7, Platform: macOS, Targets: Desktop only.

Function: Reports the absolute y coordinate of a pointing device on its tablet at full tablet resolution.

Notes: For the coordinate to be valid, the receiver should represent an event generated by a tablet pointing

device (otherwise 0 is returned). This method is valid only for mouse events with a subtype of NSTabletPointEventSubtype and for events of type NSTabletPoint. Use this value if you want to scale from tablet location to screen location yourself; otherwise use the class method `mouseLocation` or the instance method `locationInWindow`.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.22 `absoluteZ` as Integer

Plugin Version: 8.7, Platform: macOS, Targets: Desktop only.

Function: Reports the absolute z coordinate of pointing device on its tablet at full tablet resolution.

Notes: For the coordinate to be valid, the receiver should represent an event generated by a tablet pointing device (otherwise 0 is returned). The z coordinate does not represent pressure. It registers the depth coordinate returned by some tablet devices with wheels; if the device is something other than these, 0 is returned. This method is valid only for mouse events with a subtype of NSTabletPointEventSubtype and for events of type NSTabletPoint.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.23 `associatedEventsMask` as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The event mask describing the various events that you may also get during this event sequence.

Notes: This message is valid for Mouse events. Useful for determining if the input device issuing this mouse event can also simultaneously issue `NSEventTypePressure` events.

(Read only property)

4.9.24 `buttonMask` as Integer

Plugin Version: 8.7, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a bit mask identifying the buttons pressed when the tablet event represented by the receiver was generated.

Notes: Use one or more of the button-mask constants described in `Constants` to determine which buttons of the pointing device are pressed. This method is valid only for mouse events with a subtype of NSTabletPointEventSubtype and for events of type NSTabletPoint.

Available in Mac OS X v10.4 and later.

Constants to use:

<code>NSPenTipMask=1</code>	The pen tip is activated.
<code>NSPenLowerSideMask=2</code>	The button on the lower side of the device is activated.
<code>NSPenUpperSideMask=4</code>	The button on the upper side of the device is activated.

(Read only property)

4.9.25 `buttonNumber` as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the button number for the mouse button that generated an `NSOtherMouse...` event.

Notes: This method is intended for use with the `NSOtherMouseDown`, `NSOtherMouseUp`, and `NSOtherMouseDragged` events, but will return values for `NSLeftMouse...` and `NSRightMouse...` events also.

(Read only property)

4.9.26 `capabilityMask` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a mask whose set bits indicate the capabilities of the tablet device that generated the event represented by the receiver.

Notes: These bits are vendor-defined. This method is valid only for mouse events with a subtype of `NSTabletProximityEventSubtype` and for events of type `NSTabletProximity`.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.27 `CGEventRef` as Integer

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

Function: The Core Graphics event object corresponding to this event.

Notes: The `CGEventRef` opaque type returned. If no `CGEventRef` object corresponding to the `NSEvent` object can be created, this method returns 0.

(Read only property)

4.9.28 characters as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the characters associated with the receiving key-up or key-down event.

Notes: These characters are derived from a keyboard mapping that associates various key combinations with Unicode characters. Raises an `NSInternalInconsistencyException` if sent to any other kind of event object.

This method returns an empty string for dead keys, such as Option-e. However, for a key combination such as Option-Shift-e this method returns the standard accent ("").

(Read only property)

4.9.29 charactersIgnoringModifiers as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the characters generated by the receiving key event as if no modifier key (except for Shift) applies.

Notes: Raises an `NSInternalInconsistencyException` if sent to a nonkey event.

This method returns the non-modifier key character pressed for dead keys, such as Option-e. For example, Option-e (no shift key) returns an "e" for this method, whereas the `characters` method returns an empty string.

This method is useful for determining "basic" key values in a hardware-independent manner, enabling such features as keyboard equivalents defined in terms of modifier keys plus character keys. For example, to determine if the user typed Alt-S, you don't have to know whether Alt-S generates a German double ess, an integral sign, or a section symbol. You simply examine the string returned by this method along with the event's modifier flags, checking for "s" and `NSAlternateKeyMask`.

(Read only property)

4.9.30 clickCount as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the number of mouse clicks associated with the receiver, which represents a mouse-down or mouse-up event.

Notes: Raises an `NSInternalInconsistencyException` if sent to a nonmouse event.

Returns 0 for a mouse-up event if a time threshold has passed since the corresponding mouse-down event. This is because if this time threshold passes before the mouse button is released, it is no longer considered

a mouse click, but a mouse-down event followed by a mouse-up event.

The return value of this method is meaningless for events other than mouse-down or mouse-up events.
(Read only property)

4.9.31 data1 as Integer

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns additional data associated with the receiver.

Notes: The value returned by this method is dependent on the event type, and is defined by the originator of the event. Raises an `NSInternalInconsistencyException` if sent to an event not of type `NSAppKitDefined`, `NSSystemDefined`, `NSApplicationDefined`, or `NSPeriodic`.

`NSPeriodic` events don't use this attribute.
(Read only property)

4.9.32 data2 as Integer

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns additional data associated with the receiver.

Notes: The value returned by this method is dependent on the event type, and is defined by the originator of the event. Raises an `NSInternalInconsistencyException` if sent to an event not of type `NSAppKitDefined`, `NSSystemDefined`, `NSApplicationDefined`, or `NSPeriodic`.

`NSPeriodic` events don't use this attribute.
(Read only property)

4.9.33 deltaX as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the x-coordinate change for a scroll wheel, mouse-move, or mouse-drag event.

Notes: (Read only property)

4.9.34 deltaY as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the y-coordinate change for a scroll wheel, mouse-move, or mouse-drag event.

Notes: The behavior of this method may seem counter-intuitive: as the mouse moves up the screen, the value is negative; and as it moves down the screen, the value is positive. The reason for this behavior is that NSEvent computes this delta value in device space, which is flipped, but both the screen and the window's base coordinate system are not flipped.

(Read only property)

4.9.35 deltaZ as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the z-coordinate change for a scroll wheel, mouse-move, or mouse-drag event.

Notes: (Read only property)

4.9.36 description as string

Plugin Version: 11.0, Platform: macOS, Targets: Desktop only.

Function: Returns the description for this object.

Notes: (Read only property)

4.9.37 deviceID as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a special identifier that is used to match tablet-pointer events with the tablet-proximity event represented by the receiver.

Notes: All tablet-pointer events generated in the period between the device entering and leaving tablet proximity have the same device ID. This message is valid only for mouse events with subtype NSTabletPointEventSubtype or NSTabletProximityEventSubtype, and for NSTabletPoint and NSTabletProximity events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.38 eventNumber as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the counter value of the latest mouse or tracking-rectangle event object; every system-generated mouse and tracking-rectangle event increments this counter.

Notes: Raises an NSInternalInconsistencyException if sent to any other type of event object.

(Read only property)

4.9.39 Handle as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: The handle to the internal used NSEvent class.

Notes: (Read and Write property)

4.9.40 hasPreciseScrollingDeltas as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether there are precise scrolling deltas available.

Notes: Returns true if precise scrolling deltas are available; false otherwise.

This method is valid for NSScrollWheel events. A generic scroll wheel issues rather coarse scroll deltas. Some mice and trackpads provide much more precise delta. This method determines how the values of the scrollingDeltaX and scrollingDeltaY should be interpreted.

Available in OS X v10.7 and later.

(Read only property)

4.9.41 isARepeat as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns true if the receiving key event is a repeat caused by the user holding the key down, false if the key event is new.

Notes: Raises an NSInternalInconsistencyException if sent to an NSFlagsChanged event or other nonkey event.

(Read only property)

4.9.42 isDirectionInvertedFromDevice as boolean

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns whether the user has changed the device inversion.

Notes: This method valid for NSEventScrollWheel and NSEventTypeSwipe events. The user may choose to change the scrolling behavior such that it feels like they are moving the content instead of the scroll bar.

To accomplish this, `deltaX` and `deltaY` and `scrollingDeltaX` and `scrollingDeltaY` values are automatically inverted for `NSEventScrollWheel` events according to the user's preferences.

The direction of fluid swipes matches the direction of scrolling and as such for `NSEventTypeSwipe` events `gestureAmount` is inverted. However, for some uses of `NSEventScrollWheel` and `NSEventTypeSwipe` events, the behavior should not respect the user preference. This method allows you to determine when the event has been inverted and compensate by multiplying `-1` if needed.

Available in OS X v10.7 and later.
(Read only property)

4.9.43 `isEnteringProximity` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns true to indicate that a pointing device is entering the proximity of its tablet and false when it is leaving it.

Notes: This method is valid for mouse events with subtype `NSTabletProximityEventSubtype` and for `NSTabletProximity` events.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.44 `keyCode` as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns the virtual key code for the keyboard key associated with the receiving key event.

Notes: The virtual key code. The returned value is hardware-independent. The value returned is the same as the value returned in the `kEventParamKeyCode` when using Carbon Events.

Raises an `NSInternalInconsistencyException` if sent to a non-key event.
(Read only property)

4.9.45 `locationInWindow` as `NSPointMBS`

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's location in the base coordinate system of the associated window.

Notes: For nonmouse events the return value of this method is undefined.

With `NSMouseMoved` and possibly other events, the receiver can have a nil window (that is, `window` returns nil). In this case, `locationInWindow` returns the event location in screen coordinates.
(Read only property)

4.9.46 `magnification as Double`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the change in magnification.

Notes: The change in magnification that should be added to the current scaling of an item to achieve the new scale factor.

This message is valid for events of type `NSEventTypeMagnify`.

Available in Mac OS X v10.6 and later.

(Read only property)

4.9.47 `modifierFlags as Integer`

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an integer bit field indicating the modifier keys in effect for the receiver.

Notes: You can examine individual flag settings using the C bitwise AND operator with the predefined key masks. The lower 16 bits of the modifier flags are reserved for device-dependent bits.

(Read only property)

4.9.48 `pointingDeviceID as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the pointing device currently in proximity with the tablet.

Notes: This index is significant for multimode (or Dual Tracking) tablets that support multiple concurrent pointing devices; the index is incremented for each pointing device that comes into proximity. Otherwise, zero is always returned. The receiver of this message should be a mouse event object with subtype `NSTabletProximityEventSubtype` or an event of type `NSTabletProximity`.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.49 `pointingDeviceSerialNumber as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the vendor-assigned serial number of a pointing device of a certain type.

Notes: Devices of different types, such as a puck and a pen, may have the same serial number. The receiver of this message should be a mouse event object with subtype NSTabletProximityEventSubtype or an event of type NSTabletProximity.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.50 pointingDeviceType as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a NSPointingDeviceType constant indicating the kind of pointing device associated with the receiver.

Notes: For example, the device could be a pen, eraser, or cursor pointing device. This method is valid for mouse events with subtype NSTabletProximityEventSubtype and for NSTabletProximity events. See "Constants" for descriptions of valid NSPointingDeviceType constants.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.51 pressure as Double

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns a value from 0.0 through 1.0 indicating the pressure applied to the input device (used for appropriate devices).

Notes: For devices that aren't pressure-sensitive, the value is either 0.0 or 1.0. Raises an NSInternalInconsistencyException if sent to a nonmouse event.

For tablet pointing devices that are in proximity, the pressure value is 0.0 if they are not actually touching the tablet. As the device is pressed into the tablet, the value is increased.

(Read only property)

4.9.52 rotation as Double

Plugin Version: 8.7, Platform: macOS, Targets: Desktop only.

Function: Returns the rotation in degrees of the tablet pointing device associated with the receiver.

Notes: Many devices do not support rotation, in which case the returned value is 0.0. This method is valid

only for mouse events with subtype `NSTabletPointEventSubtype` and for `NSTabletPoint` events.

Available in Mac OS X v10.4 and later.
(Read only property)

4.9.53 `scrollingDeltaX` as Double

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the scroll wheel horizontal delta.

Notes: This is the preferred method for accessing `NSScrollWheel` delta values. When `hasPreciseScrollingDeltas` returns false, your application may wish to modify this value.

Available in OS X v10.7 and later.
(Read only property)

4.9.54 `scrollingDeltaY` as Double

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the scroll wheel vertical delta.

Notes: This is the preferred method for accessing `NSScrollWheel` delta values. When `hasPreciseScrollingDeltas` returns false, multiply the value returned by this method by the line or row height. Otherwise scroll by the returned amount.

Available in OS X v10.7 and later.
(Read only property)

4.9.55 `stage` as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The stage of pressure events.

Notes: This message is valid for `NSEventTypePressure` events. Pressure gesture events go through multiple stages.
(Read only property)

4.9.56 stageTransition as Double

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The stage transition for pressure events.

Notes: This message is valid for `NSEventTypePressure` events. Positive `stageTransition` describes approaching the next stage of the pressure gesture. Negative `stageTransition` describes approaching release of the current stage.

(Read only property)

4.9.57 subtype as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the subtype of the receiving event object.

Notes: Raises an `NSInternalInconsistencyException` if sent to an event not of type `NSAppKitDefined`, `NSSystemDefined`, `NSApplicationDefined`, or `NSPeriodic`.

`NSPeriodic` events don't use this attribute.

This method is also valid for mouse events on Mac OS X v10.4 and later. See the predefined mouse and tablet subtypes.

(Read only property)

4.9.58 systemTabletID as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the tablet device connected to the system.

Notes: If multiple tablets are connected to the system, the system-tablet ID is incremented for each subsequent one. If there is only one tablet device, its system-tablet ID is zero. The receiver of this message should be a mouse event object with subtype `NSTabletProximityEventSubtype` or an event of type `NSTabletProximity`.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.59 tabletID as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the USB model identifier of the tablet device associated with the receiver.

Notes: This method is valid for mouse events with subtype `NSTabletProximityEventSubtype` and for `NSTabletProximity` events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.60 `tangentialPressure` as `Double`

Plugin Version: 8.7, Platform: macOS, Targets: Desktop only.

Function: Reports the tangential pressure on the device that generated the event represented by the receiver.

Notes: The value returned can range from -1.0 to 1.0. Tangential pressure is also known as barrel pressure. Only some pointing devices support tangential pressure. This method is valid for mouse events with subtype `NSTabletPointEventSubtype` and for `NSTabletPoint` events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.61 `tilt` as `NSPointMBS`

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Reports the scaled tilt values of the pointing device that generated the event represented by the receiver.

Notes: The value returned can range from -1.0 to 1.0 for both axes. An x-coordinate value that is negative indicates a tilt to the left and a positive value indicates a tilt to the right; a y-coordinate value that is negative indicates a tilt to the top and a positive value indicates a tilt to the bottom. If the device is perfectly perpendicular to the table surface, the values are 0.0 for both axes. This method is valid for mouse events with subtype `NSTabletPointEventSubtype` and for `NSTabletPoint` events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.62 `timestamp` as `Double`

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the time the receiver occurred in seconds since system startup.

Notes: (Read only property)

4.9.63 trackingNumber as Integer

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns the identifier of a mouse-tracking event.

Notes: This method returns either an NSTrackingArea object or a NSTrackingRectTag constant depending on whether the event was generated from an NSTrackingArea object or a call to addTrackingRect:owner:user-Data:assumeInside:. Valid mouse-tracking methods are of types NSMouseEntered, NSMouseExited, and NSCursorUpdate. This method raises an NSInternalInconsistencyException if sent to any other type of event.

The NSTrackingArea class is new with Mac OS X version 10.5
(Read only property)

4.9.64 type as Integer

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: The type of the receiving event.

Notes: The type must be one of the following:

NSLeftMouseDown
NSLeftMouseUp
NSRightMouseDown
NSRightMouseUp
NSOtherMouseDown
NSOtherMouseUp
NSMouseMoved
NSLeftMouseDragged
NSRightMouseDragged
NSOtherMouseDragged
NSMouseEntered
NSMouseExited
NSKeyDown
NSKeyUp
NSFlagsChanged
NSAppKitDefined
NSSystemDefined
NSApplicationDefined
NSPeriodic
NSCursorUpdate
NSScrollWheel
(Read only property)

4.9.65 uniqueID as UInt64

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the unique identifier of the pointing device that generated the event represented by the receiver.

Notes: Also known as tool ID, this is a unique number recorded in the chip inside every pointing device. The unique ID makes it possible to assign a specific pointing device to a specific tablet. You can also use it to "sign" documents or to restrict access to document layers to a specific pointing device. This method is valid for mouse events with subtype NSTabletProximityEventSubtype and for NSTabletProximity events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.66 vendorID as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the vendor identifier of the tablet associated with the receiver.

Notes: The tablet is typically a USB device. This method is valid only for mouse events with subtype NSTabletProximityEventSubtype and for NSTabletProximity events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.67 vendorPointingDeviceType as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a coded bit field whose set bits indicate the type of pointing device (within a vendor selection) associated with the receiver.

Notes: See the vendor documentation for an interpretation of significant bits. This method is valid only for mouse events with subtype NSTabletProximityEventSubtype and for NSTabletProximity events.

Available in Mac OS X v10.4 and later.

(Read only property)

4.9.68 window as NSWindowMBS

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns the window object associated with the receiver.

Notes: A periodic event, however, has no window; in this case the return value is undefined.

(Read only property)

4.9.69 windowNumber as Integer

Plugin Version: 9.1, Platform: macOS, Targets: Desktop only.

Function: Returns the identifier for the window device associated with the receiver.

Notes: A periodic event, however, has no window; in this case the return value is undefined.

(Read only property)

4.9.70 Constants

Constants

Constant	Value	Description
NSAlphaShiftKeyMask	&h10000	One of the constants representing device-independent bits for modifier flags. Set if Caps Lock key is pressed.
NSAlternateKeyMask	&h80000	One of the constants representing device-independent bits for modifier flags. Set if Option or Alternate key is pressed.
NSAnyEventMask	&hFFFFFFFF	One of the constants for masks for the events. Pass them to the NSCell method <code>sendActionOn</code> to specify which events should send its action message.
NSAppKitDefined	13	One of the event types.
NSAppKitDefinedMask	&h2000	One of the constants for masks for the events. Pass them to the NSCell method <code>sendActionOn</code> to specify which events should send its action message.
NSApplicationActivatedEventType	1	One of the constants representing the types of events defined by the NSAppKit. The application has been activated.
NSApplicationDeactivatedEventType	2	One of the constants representing the types of events defined by the NSAppKit. The application has been deactivated.
NSApplicationDefined	15	One of the event types.
NSApplicationDefinedMask	&h8000	One of the constants for masks for the events. Pass them to the NSCell method <code>sendActionOn</code> to specify which events should send its action message.
NSAWTEventType	16	One of the constants representing the types of events defined by the NSAppKit.
NSBeginFunctionKey	&hF72A	An event type used to support Java applications. One of the constants representing Unicode characters (0xF72A-0xF72F) are reserved for function keys on the keyboard. Begin key. Not on most Macintosh keyboards.
NSBreakFunctionKey	&hF732	One of the constants representing Unicode characters (0xF732-0xF737) are reserved for function keys on the keyboard. Break key. Not on most Macintosh keyboards.
NSClearDisplayFunctionKey	&hF73A	One of the constants representing Unicode characters (0xF73A-0xF73F) are reserved for function keys on the keyboard. Clear Display key. Not on most Macintosh keyboards.
NSClearLineFunctionKey	&hF739	One of the constants representing Unicode characters (0xF739-0xF73E) are reserved for function keys on the keyboard. Clear/Num Lock key.
NSCommandKeyMask	&h100000	One of the constants representing device-independent bits for modifier flags. Set if Command key is pressed.
NSControlKeyMask	&h40000	One of the constants representing device-independent bits for modifier flags. Set if Control key is pressed.
NSCursorPointingDevice	2	One of the constants represent pointing-device types for NSEvent events or mouse events with subtype NSTabletProximityEvent. The <code>pointingDeviceType</code> method returns one of these constants. Represents a cursor (or puck-like) pointing device. Available in Mac OS X v10.4 and later.
NSCursorUpdate	17	One of the event types.
NSCursorUpdateMask	&h20000	One of the constants for masks for the events. Pass them to the NSCell method <code>sendActionOn</code> to specify which events should send its action message.
NSDeleteCharFunctionKey	&hF73E	One of the constants representing Unicode characters (0xF73E-0xF741) are reserved for function keys on the keyboard. Delete Character key. Not on most Macintosh keyboards.
NSDeleteFunctionKey	&hF728	One of the constants representing Unicode characters (0xF728-0xF72F) are reserved for function keys on the keyboard.

4.10 class NSFontDescriptorMBS

4.10.1 class NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: The Cocoa class for a font descriptor.

Notes: NSFontDescriptor objects provide a mechanism to describe a font with a dictionary of attributes. This font descriptor can be used later to create or modify an NSFont object. Mac OS X v10.4 and later provides a font matching capability, so that you can partially describe a font by creating a font descriptor with, for example, just a family name. You can then find all the available fonts on the system with a matching family name using `matchingFontDescriptorsWithMandatoryKeys`.

There are several ways to create a new NSFontDescriptor object. You can use the Constructor, `fontDescriptorWithFontAttributes` or `fontDescriptorWithName` to create a font descriptor based on either your custom attributes dictionary or on a specific font's name and size. Alternatively you can use one of the `fontDescriptor` instance methods (such as `fontDescriptorWithFace`) to create a modified version of an existing descriptor. The latter methods are useful if you have an existing descriptor and simply want to change one aspect.

All attributes in the attributes dictionary are optional.

Blog Entries

- [CocoaBase needs MacOSXCG](#)
- [MBS Real Studio Plugins, version 12.1pr10](#)

4.10.2 Methods

4.10.3 Constructor(AttributesDic as Dictionary)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Initializes and returns a new font descriptor with the specified attributes.

Notes: `attributes`: The attributes for the new font descriptor. If nil, the font descriptor's attribute dictionary will be empty.

4.10.4 copy as NSFontDescriptorMBS

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

Function: Creates a copy of the object.

4.10.5 fontAttributes as Dictionary

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the receiver's dictionary of attributes.

4.10.6 fontDescriptorByAddingAttributes(AttributesDic as Dictionary) as NS-FontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor that is the same as the receiver but with the specified attributes taking precedence over the existing ones.

4.10.7 fontDescriptorWithFace(newFace as string) as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor that is the same as the receiver but with the specified face.

Notes: newFace: The new font face.

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

4.10.8 fontDescriptorWithFamily(newFamily as string) as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor whose attributes are the same as the receiver but from the specified family.

Notes: newFamily: The new font family.

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

4.10.9 fontDescriptorWithFontAttributes(AttributesDic as Dictionary) as NS-FontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a font descriptor with a dictionary of attributes.

Example:

```
// ask for monospace font trait
Dim traitsAttributes As New Dictionary
traitsAttributes.Value(NSFontDescriptorMBS.NSFontSymbolicTrait) = NSFontDescriptorMBS.NSFontMonoSpace-
Trait

// ask for traits
Dim fontAttributes As New Dictionary
fontAttributes.Value(NSFontDescriptorMBS.NSFontTraitsAttribute) = traitsAttributes

// now make a font descriptor for this
Dim fd As NSFontDescriptorMBS = NSFontDescriptorMBS.fontDescriptorWithFontAttributes(fontAttributes)

// and ask font manager for matching fonts
Dim fontManager As New NSFontManagerMBS
Dim fonts() As String = fontManager.availableFontNamesMatchingFontDescriptor(fd)

// finds e.g. AndaleMono, CourierNewPSMT, CourierNewPS-ItalicMT, CourierNewPS-BoldMT, CourierNewPS-
BoldItalicMT,
// FZLTXHB-B51-0, FZLTZHB-B51-0, FZLTTHB-B51-0, Menlo-Regular, Menlo-Italic, Menlo-Bold, Menlo-
BoldItalic,
// Monaco, Osaka-Mono, JCsmPC, PTMono-Regular, PTMono-Bold
```

Notes: attributes: The attributes for the font descriptor. If nil, the font descriptor's dictionary will be empty.

Returns the new font descriptor.

4.10.10 fontDescriptorWithMatrix(matrix as Variant) as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor that is the same as the receiver but with the specified matrix.

Notes: matrix: The new font matrix. Must be a NSAffineTransformMBS object.

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

4.10.11 fontDescriptorWithName(fontName as string, matrix as Variant) as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a font descriptor with the `NSFontNameAttribute` and `NSFontMatrixAttribute` dictionary attributes set to the given values.

Notes: `fontName`: The value for `NSFontNameAttribute`.

`matrix`: The value for `NSFontMatrixAttribute`. Must be a `NSAffineTransformMBS` object or `nil`!

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

See also:

- 4.10.12 `fontDescriptorWithName(fontName as string, size as Double)` as `NSFontDescriptorMBS` 252

4.10.12 `fontDescriptorWithName(fontName as string, size as Double)` as `NSFontDescriptorMBS`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a font descriptor with the `NSFontNameAttribute` and `NSFontSizeAttribute` dictionary attributes set to the given values.

Notes: `fontName`: The value for `NSFontNameAttribute`.

`size`: The value for `NSFontSizeAttribute`.

Returns the new font descriptor.

Available in Mac OS X v10.3 and later.

See also:

- 4.10.11 `fontDescriptorWithName(fontName as string, matrix as Variant)` as `NSFontDescriptorMBS` 251

4.10.13 `fontDescriptorWithSize(newPointSize as Double)` as `NSFontDescriptorMBS`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor that is the same as the receiver but with the specified point size.

Notes: `newPointSize`: The new point size.

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

4.10.14 fontDescriptorWithSymbolicTraits(SymbolicTraits as Integer) as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a new font descriptor that is the same as the receiver but with the specified symbolic traits taking precedence over the existing ones.

Example:

```
// get a bold font
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(13)
dim f as NSFontDescriptorMBS = n.fontDescriptor
dim w as Integer = f.symbolicTraits

// find similar font without bold
f = f.fontDescriptorWithSymbolicTraits( BitwiseAnd(w, Bitwise.OnesComplement(f.NSFontBoldTrait)))

// now check it
dim d as Dictionary = f.fontAttributes
w = f.symbolicTraits

if BitwiseAnd(w, f.NSFontBoldTrait) <> 0 then
  MsgBox "bold"
else
  MsgBox "not bold"
end if
```

Notes: symbolicTraits: The new symbolic traits.

Returns the new font descriptor.

Available in Mac OS X v10.4 and later.

4.10.15 matchingFontDescriptorsWithMandatoryKeys as NSFontDescriptorMBS()

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns all the fonts available on the system whose specified attributes match those of the receiver.

Notes: mandatoryKeys: Optional, keys that must be identical to be matched. Can be nil.

Returns the matching font descriptors.

For example, suppose there are two versions of a given font installed that differ in the number of glyphs

covered (the new version has more glyphs). If you explicitly specify `NSFontNameAttribute` as the only mandatory key, then a font descriptor that specifies a font name and character set by default matches both versions, since the character set attribute is not used for matching. If you specify that font name and character set keys are mandatory, the returned array contains only the font that matches both keys.

Available in Mac OS X v10.4 and later.
See also:

- 4.10.16 `matchingFontDescriptorsWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS()` 254

4.10.16 `matchingFontDescriptorsWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS()`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns all the fonts available on the system whose specified attributes match those of the receiver.

Notes: `mandatoryKeys`: Optional, keys that must be identical to be matched. Can be nil.

Returns the matching font descriptors.

For example, suppose there are two versions of a given font installed that differ in the number of glyphs covered (the new version has more glyphs). If you explicitly specify `NSFontNameAttribute` as the only mandatory key, then a font descriptor that specifies a font name and character set by default matches both versions, since the character set attribute is not used for matching. If you specify that font name and character set keys are mandatory, the returned array contains only the font that matches both keys.

Available in Mac OS X v10.4 and later.
See also:

- 4.10.15 `matchingFontDescriptorsWithMandatoryKeys as NSFontDescriptorMBS()` 253

4.10.17 `matchingFontDescriptorWithMandatoryKeys as NSFontDescriptorMBS`

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

Function: Returns a normalized font descriptor whose specified attributes match those of the receiver.

Notes: `mandatoryKeys`: Optional, keys that must be identical to be matched. Can be nil.

Returns the matching font descriptor.

The returned font descriptor is the first element returned from `matchingFontDescriptorsWithMandatoryKeys`.

See also:

- 4.10.18 `matchingFontDescriptorWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS` 255

4.10.18 `matchingFontDescriptorWithMandatoryKeys(mandatoryKeys() as string) as NSFontDescriptorMBS`

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

Function: Returns a normalized font descriptor whose specified attributes match those of the receiver.

Notes: `mandatoryKeys`: Optional, keys that must be identical to be matched. Can be nil.

Returns the matching font descriptor.

The returned font descriptor is the first element returned from `matchingFontDescriptorsWithMandatoryKeys`.

See also:

- 4.10.17 `matchingFontDescriptorWithMandatoryKeys as NSFontDescriptorMBS` 254

4.10.19 `matrix as Variant`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the current transform matrix of the receiver.

Notes: Must be a `NSAffineTransformMBS` object or nil!

4.10.20 `NSFontCascadeListAttribute as string`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An array with sub-descriptors.

The default value is the system default cascading list for user's locale.

Available in Mac OS X v10.4 and later.

4.10.21 `NSFontCharacterSetAttribute as string`

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An `NSCharacterSetMBS` instance that represents the set of Unicode characters covered by the font.

The default value is supplied by the font.

Available in Mac OS X v10.4 and later.

4.10.22 NSFontColorAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional memoryblock that specifies the font color. (Deprecated. Use NSForegroundColorAttributeName instead.)

Available in Mac OS X v10.3 and later.

Deprecated in Mac OS X v10.4.

4.10.23 NSFontFaceAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional string that specifies the font face.

4.10.24 NSFontFamilyAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional string that specifies the font family.

4.10.25 NSFontFeatureSelectorIdentifierKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys for font descriptor attributes.

Notes: A number specifying a font feature selector such as common ligature off, traditional character shape, and so on. See "Font Features" in ATSUI Programming Guide for predefined feature selectors.

Available in Mac OS X v10.5 and later.

4.10.26 NSFontFeatureTypeIdentifierKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys for font descriptor attributes.

Notes: A number specifying a font feature type such as ligature, character shape, and so on. See "Font Features" in ATSUI Programming Guide for predefined feature types.

Available in Mac OS X v10.5 and later.

4.10.27 NSFontFixedAdvanceAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: A number containing a float value that overrides the glyph advancement specified by the font. The default value is 0.0.

4.10.28 NSFontMatrixAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An NSAffineTransformMBS instance that specifies the font's transformation matrix. The default value is the identity matrix. Available in Mac OS X v10.4 and later.

4.10.29 NSFontNameAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional string that specifies the font name.

4.10.30 NSFontSizeAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional string, containing a float value, that specifies the font size.

4.10.31 NSFontSlantTrait as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its trait dictionary.

Notes: The relative slant angle value as a number.

The valid value range is from -1.0 to 1.0. The value of 0.0 corresponds to 0 degree clockwise rotation from the vertical and 1.0 corresponds to 30 degrees clockwise rotation.

Available in Mac OS X v10.4 and later.

4.10.32 NSFontSymbolicTrait as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its trait dictionary.

Notes: The symbolic traits value as a number.

Available in Mac OS X v10.4 and later.

4.10.33 NSFontTraitsAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: A Dictionary instance instance fully describing font traits.

The default value is supplied by the font. See "Font traits dictionary keys" for dictionary keys.

Available in Mac OS X v10.4 and later.

4.10.34 NSFontVariationAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: A dictionary instance that describes the font's variation axis.

The default value is supplied by the font. See "Font variation axis dictionary keys" for dictionary keys.

Available in Mac OS X v10.4 and later.

4.10.35 NSFontVariationAxisDefaultValueKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its variation axis dictionary.

Notes: The default axis value as a number.

Available in Mac OS X v10.4 and later.

4.10.36 NSFontVariationAxisIdentifierKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its variation axis dictionary.

Notes: The axis identifier value as a number.

Available in Mac OS X v10.4 and later.

4.10.37 NSFontVariationAxisMaximumValueKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its variation axis dictionary.

Notes: The maximum axis value as a number.

Available in Mac OS X v10.4 and later.

4.10.38 NSFontVariationAxisMinimumValueKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its variation axis dictionary.

Notes: The minimum axis value as a number.

Available in Mac OS X v10.4 and later.

4.10.39 NSFontVariationAxisNameKey as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its variation axis dictionary.

Notes: The localized variation axis name.

Available in Mac OS X v10.4 and later.

4.10.40 NSFontVisibleNameAttribute as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the attribute keys.

Notes: An optional string that specifies the font's visible name.

4.10.41 NSFontWeightTrait as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its trait dictionary.

Notes: The normalized weight value as a number.

The valid value range is from -1.0 to 1.0. The value of 0.0 corresponds to the regular or medium font weight.

Available in Mac OS X v10.4 and later.

4.10.42 NSFontWidthTrait as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: One of the keys to retrieve information about a font descriptor from its trait dictionary.

Notes: The relative inter-glyph spacing value as a number.

The valid value range is from -1.0 to 1.0. The value of 0.0 corresponds to the regular glyph spacing. Available in Mac OS X v10.4 and later.

4.10.43 pointSize as Double

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the point size of the receiver.

Example:

```
dim f as NSFontMBS = NSFontMBS.fontWithName("Arial", 12)
dim d as NSFontDescriptorMBS = f.fontDescriptor
MsgBox str(d.pointSize)
```

4.10.44 postscriptName as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the PostScript name of the receiver.

Example:

```
dim f as NSFontMBS = NSFontMBS.fontWithName("Arial", 12)
dim d as NSFontDescriptorMBS = f.fontDescriptor
MsgBox d.postscriptName
```

4.10.45 symbolicTraits as Integer

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a bit mask that describes the traits of the receiver.

Example:

```
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(13)
dim f as NSFontDescriptorMBS = n.fontDescriptor
dim w as Integer = f.symbolicTraits
```

```

if BitwiseAnd(w, f.NSFontBoldTrait) <>0 then
MsgBox "bold"
else
MsgBox "not bold"
end if

```

Notes: Typeface information is specified by the lower 16 bits of system trait value.

The font family class constants classify certain stylistic qualities of the font. These values correspond closely to the font class values in the OpenType OS/2 table. The class values are bundled in the upper four bits of the NSFontSymbolicTraits and can be accessed via NSFontFamilyClassMask. For more information about the specific meaning of each identifier, refer to the OpenType specification.

4.10.46 variantForKey(key as string) as Variant

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the font attribute specified by the given key.

4.10.47 Properties

4.10.48 Handle as Integer

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: The internal reference.

Notes: (Read and Write property)

4.10.49 Constants

Constants

Constant	Value	Description
NSFontFamilyClassMask	&hF0000000	The font family class mask used to access NSFontFamilyClass values. Available in Mac OS X v10.4 and later. This constant is used to access NSFontFamilyClass values in the upper four bits of NSFontSymbolicTraits.

Typeface Constants

Constant	Value	Description
NSFontBoldTrait	2	The font's typestyle is boldface. Available in Mac OS X v10.4 and later.
NSFontCondensedTrait	64	The font's typestyle is condensed. Expanded and condensed traits are mutually exclusive. Available in Mac OS X v10.4 and later.
NSFontExpandedTrait	32	The font's typestyle is expanded. Expanded and condensed traits are mutually exclusive. Available in Mac OS X v10.4 and later.
NSFontItalicTrait	1	The font's typestyle is italic. Available in Mac OS X v10.4 and later.
NSFontMonoSpaceTrait	1024	The font uses fixed-pitch glyphs if available. The font may have multiple glyph advances (many CJK glyphs contain two spaces). Available in Mac OS X v10.4 and later.
NSFontUIOptimizedTrait	4096	The font synthesizes appropriate attributes for user interface rendering, such as control titles, if necessary. Available in Mac OS X v10.4 and later.
NSFontVerticalTrait	2048	The font uses vertical glyph variants and metrics. Available in Mac OS X v10.4 and later.

Font Family Class Constants

Constant	Value	Description
NSFontClarendonSerifsClass	&h40000000	The font's style is a variation of the Oldstyle Serifs and the Transitional Serifs. Available in Mac OS X v10.4 and later.
NSFontFreeformSerifsClass	&h70000000	The font's style includes serifs, but it expresses a design freedom that does not generally fit within the other serif design classifications. Available in Mac OS X v10.4 and later.
NSFontModernSerifsClass	&h30000000	The font's style is based on the Latin printing style of the 20th century. Available in Mac OS X v10.4 and later.
NSFontOldStyleSerifsClass	&h10000000	The font's style is based on the Latin printing style of the 15th to 17th centuries. Available in Mac OS X v10.4 and later.
NSFontOrnamentalsClass	&h90000000	The font's style includes highly decorated or stylized character shapes such as those typically used in headlines. Available in Mac OS X v10.4 and later.
NSFontSansSerifClass	&h80000000	The font's style includes most basic letter forms (excluding Scripts and Ornamentals) that do not have serifs on the strokes. Available in Mac OS X v10.4 and later.
NSFontScriptsClass	&hA0000000	The font's style is among those typefaces designed to simulate handwriting. Available in Mac OS X v10.4 and later.
NSFontSlabSerifsClass	&h50000000	The font's style is characterized by serifs with a square transition between the strokes and the serifs (no brackets). Available in Mac OS X v10.4 and later.
NSFontSymbolicClass	&hC0000000	The font's style is generally design independent, making it suitable for special characters (icons, dingbats, technical symbols, and so on) that may be used equally well with any font. Available in Mac OS X v10.4 and later.
NSFontTransitionalSerifsClass	&h20000000	The font's style is based on the Latin printing style of the 18th to 19th centuries. Available in Mac OS X v10.4 and later.
NSFontUnknownClass	0	The font has no design classification. Available in Mac OS X v10.4 and later.

4.11 class NSFontMBS

4.11.1 class NSFontMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: A class for a font in the Cocoa world.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.4](#)
- [MBS Xojo Plugins in version 22.4](#)
- [MBS Xojo Plugins, version 22.4pr5](#)
- [Styled Text for Labels in your Xojo iOS app](#)
- [Apply fonts to font PopupMenu](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr2](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)
- [CocoaBase needs MacOSXCG](#)
- [MBS Real Studio Plugins, version 12.1pr10](#)
- [MBS REALbasic Plugins, version 10.6pr2](#)

Xojo Developer Magazine

- [6.1, page 6: Thoughts from the Publisher by Marc Zeedar](#)
- [20.6, page 9: News](#)

4.11.2 Methods

4.11.3 advancementForGlyph(aGlyph as Integer) as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the nominal spacing for the given glyph—the distance the current point moves after showing the glyph—accounting for the receiver's size.

Notes: This spacing is given according to the glyph's movement direction, which is either strictly horizontal or strictly vertical.

4.11.4 boldSystemFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the Aqua system font used for standard interface items that are rendered in boldface type in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.boldSystemFontOfSize(12)
```

MsgBox n.description

```
// shows: "LucidaGrande-Bold 12.00 pt. P [ ] (0x1c082c0) fobj=0x1c09d30, spc=3.96"
```

Notes: If `fontSize` is 0 or negative, returns the boldface system font at the default size.

4.11.5 boundingRectForGlyph(aGlyph as Integer) as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the bounding rectangle for the specified glyph, scaled to the receiver's size.

Notes: Japanese fonts encoded with the scheme "EUC12-NJE-CFEncoding" do not have individual metrics or bounding boxes available for the glyphs above 127. For those glyphs, this method returns the bounding rectangle for the font instead.

4.11.6 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

4.11.7 controlContentFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for the content of controls in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.controlContentFontOfSize(12)
```

MsgBox n.description

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x16017750) fobj=0x160175c0, spc=3.80"
```

Notes: For example, in a table, the user's input uses the control content font, and the table's header uses another font. If `fontSize` is 0 or negative, returns the control content font at the default size.

4.11.8 file as folderitem

Plugin Version: 13.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the folderitem for this font.

Notes: One file may give several font faces.

4.11.9 fontDescriptor as NSFontDescriptorMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns the receiver's font descriptor.

Notes: A font descriptor object that describes the receiver.

The font descriptor contains a mutable dictionary of optional attributes for creating an `NSFont` object. See documentation on `NSFontDescriptor` for more information.

4.11.10 fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, fontSize as Double) as NSFontMBS

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Returns a font object for the specified font descriptor and font size.

Notes: `fontDescriptor`: A font descriptor object.

`fontSize`: The size in points to which the font is scaled.

Returns a font object for the specified descriptor and size.

In most cases, you can simply use `fontWithName` to create standard scaled fonts.

Available in Mac OS X v10.4 and later.

See also:

- 4.11.11 `fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, TextTransform as Variant) as NSFontMBS` 267

4.11.11 fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, TextTransform as Variant) as NSFontMBS

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

Function: Returns a font object for the specified font descriptor and text transform.

Notes: fontDescriptor: The font descriptor object describing the font to return.

textTransform: An affine transformation applied to the font. Must be a NSAffineTransformMBS object or nil!

Returns a font object for the specified name and transform.

In most cases, you can simply use fontWithName to create standard scaled fonts. If textTransform is non-nil, it has precedence over NSFontMatrixAttribute in fontDescriptor.

Available in Mac OS X v10.4 and later.

See also:

- 4.11.10 fontWithDescriptor(fontDescriptor as NSFontDescriptorMBS, fontSize as Double) as NSFontMBS
266

4.11.12 fontWithName(fontName as string, fontSize as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Creates a font object for the specified font name and font size.

Example:

```
dim n as NSFontMBS = NSFontMBS.fontWithName("Monaco", 12)
```

MsgBox n.description

```
// shows: "Monaco 12.00 pt. P [ ] (0x1805020) fobj=0x168114b0, spc=7.20"
```

Notes: The fontName is a fully specified family-face name, such as Helvetica-BoldOblique or Times-Roman. The fontSize is equivalent to using a font matrix of [fontSize 0 0 fontSize 0 0]. If you use a fontSize of 0.0, this method uses the default User Font size.

Fonts created with this method automatically flip themselves in flipped views. This method is the preferred means for creating fonts.

4.11.13 glyphWithName(name as string) as UInt32

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the named encoded glyph, or -1 if the receiver contains no such glyph.

Notes: Returns -1 if the glyph named glyphName isn't encoded.

Glyph names in fonts do not always accurately identify the glyph. If possible, look up the appropriate glyph on your own.

4.11.14 labelFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the Aqua font used for standard interface labels in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.labelFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x1829350) fobj=0x18293f0, spc=3.80"
```

Notes: If fontSize is 0 or negative, returns the label font with the default size.

4.11.15 labelFontSize as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the size of the standard label font.

Example:

```
MsgBox str(NSFontMBS.labelFontSize) // shows 10
```

4.11.16 menuBarFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for menu bar items, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.menuBarFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x17e00c00) fobj=0x17e00ca0, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the menu bar font with the default size. Available in Mac OS X v10.3 and later.

4.11.17 `menuFontOfSize(size as Double)` as `NSFontMBS`

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for menu items, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.menuFontOfSize(12)
```

MsgBox n.description

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x1da6d80) fobj=0x1da6e20, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the menu items font with the default size.

4.11.18 `messageFontOfSize(size as Double)` as `NSFontMBS`

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for standard interface items, such as button labels, menu items, and so on, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.messageFontOfSize(12)
```

MsgBox n.description

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x1c11520) fobj=0x1c11790, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns this font at the default size. This method is equivalent to `systemFontOfSize()`.

4.11.19 `monospacedDigitSystemFontOfSize(fontSize as double, weight as double)` as `NSFontMBS`

Plugin Version: 22.4, Platform: macOS, Targets: All.

Function: Returns a version of the standard system font that contains monospaced digit glyphs.

Notes: `fontSize`: The desired font size specified in points. If you specify 0.0 or a negative number for this

parameter, the method returns the system font at the default size.

weight: The desired weight of font lines, specified as one of the constants in `NSFontWeight`.

Returns a font object containing the system font with monospace digits at the specified size and weight.

The font returned by this method has monospaced digit glyphs. Glyphs for other characters and symbols may be wider or narrower than the monospaced characters. To ensure the font uses fixed spacing for all characters, apply the `NSFontFixedAdvanceAttribute` attribute to the any strings you render.

4.11.20 `monospacedSystemFontOfSize(fontSize as double, weight as double) as NSFontMBS`

Plugin Version: 22.4, Platform: macOS, Targets: All.

Function: Returns a monospace version of the system font with the specified size and weight.

Notes: `fontSize`: The desired font size specified in points. If you specify 0.0 or a negative number for this parameter, the method returns the system font at the default size.

weight: The desired weight of font lines, specified as one of the constants in `NSFontWeight`.

Returns a font object containing a monospace version of the system font at the specified size and weight.

Use the returned font for interface items that require monospaced glyphs. The returned font includes monospaced glyphs for the Latin characters and the symbols commonly found in source code. Glyphs for other symbols are usually wider or narrower than the monospaced characters. To ensure the font uses fixed spacing for all characters, apply the `NSFontFixedAdvanceAttribute` attribute to the any strings you render.

4.11.21 `paletteFontOfSize(size as Double) as NSFontMBS`

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for palette window title bars, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.paletteFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x18800c00) fobj=0x18800ca0, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the palette title font at the default size.

4.11.22 `screenFontWithRenderingMode(renderingMode as Integer) as NSFontMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a bitmapped screen font, when sent to a font object representing a scalable PostScript font, with the specified rendering mode, matching the receiver in typeface and matrix (or size), or nil if such a font can't be found.

Notes: For valid rendering modes, see `NSFontRenderingMode`.

Screen fonts are for direct use with the window server only. Never use them with Application Kit objects, such as in `setFont:` methods. Internally, the Application Kit automatically uses the corresponding screen font for a font object as long as the view is not rotated or scaled.

Available in Mac OS X v10.4 and later.

4.11.23 `setUserFixedPitchFont(font as NSFontMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the font used by default for documents and other text under the user's control, when that font should be fixed-pitch, to the specified font.

Notes: Specifying font as nil causes the default to be removed from the application domain.

4.11.24 `setUserFont(font as NSFontMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the font used by default for documents and other text under the user's control to the specified font.

Notes: Specifying font as nil causes the default to be removed from the application domain.

4.11.25 `smallSystemFontSize as Double`

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the size of the standard small system font.

Example:

```
MsgBox str(NSFontMBS.smallSystemFontSize) // shows 11
```

4.11.26 `systemFontSize(size as Double) as NSFontMBS`

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the Aqua system font used for standard interface items, such as button labels, menu items, and so on, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.systemFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x17e009d0) fobj=0x17e00ca0, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the system font at the default size.

4.11.27 `systemFontSize` as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the size of the standard system font.

Example:

```
MsgBox str(NSFontMBS.systemFontSize) // shows 13
```

4.11.28 `systemFontSizeForControlSize(controlSize as Integer)` as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font size used for the specified control size.

Example:

```
MsgBox str(NSFontMBS.systemFontSizeForControlSize(0)) // 13
```

```
MsgBox str(NSFontMBS.systemFontSizeForControlSize(1)) // 11
```

Notes: If `controlSize` does not correspond to a valid `NSControlSize`, returns the size of the standard system font.

Available in Mac OS X v10.3 and later.

4.11.29 `titleBarFontOfSize(size as Double)` as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for window title bars, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.titleBarFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x1a31b20) fobj=0x1a31bc0, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the title bar font at the default size. This method is equivalent to `boldSystemFontOfSize`.

4.11.30 toolTipsFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used for tool tips labels, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.toolTipsFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "LucidaGrande 12.00 pt. P [ ] (0x1828db0) fobj=0x1828e50, spc=3.80"
```

Notes: If `fontSize` is 0 or negative, returns the tool tips font at the default size.

4.11.31 userFixedPitchFontOfSize(size as Double) as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used by default for documents and other text under the user's control (that is, text whose font the user can normally change), when that font should be fixed-pitch, in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.userFixedPitchFontOfSize(12)
```

```
MsgBox n.description
```

```
// shows: "Monaco 12.00 pt. P [ ] (0x1ad13a0) fobj=0x1ac2bd0, spc=7.20"
```

Notes: If `fontSize` is 0 or negative, returns the fixed-pitch font at the default size.

The system does not guarantee that all the glyphs in a fixed-pitch font are the same width. For example, certain Japanese fonts are dual-pitch, and other fonts may have nonspacing marks that can affect the display of other glyphs.

4.11.32 `userFontSize(size as Double)` as `NSFontMBS`

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the font used by default for documents and other text under the user's control (that is, text whose font the user can normally change), in the specified size.

Example:

```
dim n as NSFontMBS = NSFontMBS.userFontSize(12)
```

```
MsgBox n.description
```

```
// shows: "Helvetica 12.00 pt. P [ ] (0x19019b0) fobj=0x181fd00, spc=3.33"
```

Notes: If `fontSize` is 0 or negative, returns the user font at the default size.

4.11.33 Properties

4.11.34 `ascender` as `Double`

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the top y-coordinate, offset from the baseline, of the receiver's longest ascender.

Notes: (Read only property)

4.11.35 `boundingRectForFont` as `NSRectMBS`

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's bounding rectangle, scaled to the font's size.

Notes: The bounding rectangle is the union of the bounding rectangles of every glyph in the font.
(Read only property)

4.11.36 `capHeight` as `Double`

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the receiver's cap height.

Notes: (Read only property)

4.11.37 coveredCharacterSet as Variant

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSMutableCharacterSet object containing all of the nominal characters renderable by the receiver, which is all of the entries mapped in the receiver's 'Àömap' table.

Example:

```
// get the Symbol font
dim n as NSFontMBS = NSFontMBS.fontWithName("Symbol",10)

// what characters are defined for this font?
dim c as NSMutableCharacterSet = n.coveredCharacterSet

// display a string with all the characters
MsgBox c.StringValue
```

Notes: Value is a NSMutableCharacterSetMBS object. Returned as Variant to reduce plugin dependencies.

The number of glyphs supported by a given font is often larger than the number of characters contained in the character set returned by this method.

(Read only property)

4.11.38 descender as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the bottom y coordinate, offset from the baseline, of the receiver's longest descender.

Notes: Thus, if the longest descender extends 2 points below the baseline, descender will return -2.

(Read only property)

4.11.39 description as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The description of the font.

Example:

```
dim n as NSFontMBS = NSFontMBS.systemFontOfSize(12)
MsgBox n.description
```

Notes: (Read only property)

4.11.40 `displayName` as string

Plugin Version: 8.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The (localized) display name of the font.

Example:

```
dim n as NSFontMBS = NSFontMBS.fontWithName("Monaco", 12)
```

```
MsgBox n.displayName // shows "Monaco"
```

Notes: (Read only property)

4.11.41 `familyName` as string

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: The family name of the font.

Example:

```
dim n as NSFontMBS = NSFontMBS.fontWithName("Monaco", 12)
```

```
MsgBox n.familyName // shows "Monaco"
```

Notes: (Read only property)

4.11.42 `fontName` as string

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: The name of the font.

Example:

```
dim n as NSFontMBS = NSFontMBS.fontWithName("Monaco", 12)
```

```
MsgBox n.fontName // shows "Monaco"
```

Notes: (Read only property)

4.11.43 Handle as Integer

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: The internal reference to the NSFont object.

Notes: (Read and Write property)

4.11.44 isFixedPitch as boolean

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether all glyphs in the receiver have the same advancement.

Notes: Returns true if all glyphs in the receiver have the same advancement; false if any advancements differ.

Some Japanese fonts encoded with the scheme "EUC12-NJE-CFEncoding" return that they have the same advancement, but actually encode glyphs with one of two advancements, for historical compatibility. You may need to handle such fonts specially for some applications.

(Read only property)

4.11.45 italicAngle as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's italic angle, the amount that the font is slanted in degrees counterclockwise from the vertical, as read from its AFM file.

Notes: Because the slant is measured counterclockwise, English italic fonts typically return a negative value.
(Read only property)

4.11.46 leading as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the receiver's leading.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

4.11.47 maximumAdvancement as NSSizeMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the greatest advancement of any of the receiver’s glyphs.

Notes: This advancement is always either strictly horizontal or strictly vertical.
(Read only property)

4.11.48 mostCompatibleStringEncoding as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the string encoding that works best with the receiver, where there are the fewest possible unmatched characters in the string encoding and glyphs in the font.

Notes: You can use NSString’s `dataUsingEncoding` method to convert the string to this encoding. If this method returns `NSASCIIStringEncoding`, it could not determine the correct encoding and assumed that the font can render only ASCII characters.

This method works heuristically using well-known font encodings, so for nonstandard encodings it may not in fact return the optimal string encoding.

(Read only property)

4.11.49 numberOfGlyphs as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of glyphs in the receiver.

Example:

```
dim n as NSFontMBS = NSFontMBS.fontWithName("Monaco", 12)
```

```
MsgBox str(n.numberOfGlyphs) // 1678
```

Notes: Glyphs are numbered starting at 0.

(Read only property)

4.11.50 pointSize as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the receiver’s point size, or the effective vertical point size for a font with a nonstandard matrix.

Notes: (Read only property)

4.11.51 printerFont as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the scalable PostScript font corresponding to itself.

Notes: When sent to a font object representing a scalable PostScript font, returns self. When sent to a font object representing a bitmapped screen font, returns its corresponding scalable PostScript font.

(Read only property)

4.11.52 renderingMode as Integer

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the rendering mode of the receiver.

Notes: Available in Mac OS X v10.4 and later.

(Read only property)

4.11.53 screenFont as NSFontMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the bitmapped screen font corresponding to itself.

Notes: When sent to a font object representing a scalable PostScript font, returns a bitmapped screen font matching the receiver in typeface and matrix (or size), or nil if such a font can't be found. When sent to a font object representing a bitmapped screen font, returns nil.

Screen fonts are for direct use with the window server only. Never use them with Application Kit objects, such as in setFont: methods. Internally, the Application Kit automatically uses the corresponding screen font for a font object as long as the view is not rotated or scaled.

(Read only property)

4.11.54 textTransform as Variant

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the current transformation matrix for the receiver.

Notes: Available in Mac OS X v10.4 and later.

Returns NSAffineTransformMBS. Returned as Variant to reduce plugin dependencies.

(Read only property)

4.11.55 underlinePosition as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the baseline offset that should be used when drawing underlines with the receiver, as determined by the font's AFM file.

Notes: This value is usually negative, which must be considered when drawing in a flipped coordinate system.

(Read only property)

4.11.56 underlineThickness as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the thickness that should be used when drawing underlines with the receiver, as determined by the font's AFM file.

Notes: (Read only property)

4.11.57 xHeight as Double

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Returns the x-height of the font.

Notes: (Read only property)

4.11.58 Constants

Constants

Constant	Value	Description
NSControlGlyph	&hFFFFFF	One of the constants define reserved glyph codes. NSGlyphGenerator generates NSControlGlyph for General Category C* and U200B (ZERO WIDTH SPACE).
NSFontAntialiasedIntegerAdvancementsRenderingMode	3	One of the constants specify the font rendering mode. Specifies antialiased, integer advancements rendering mode. Available in Mac OS X v10.4 and later.
NSFontAntialiasedRenderingMode	1	One of the constants specify the font rendering mode. Specifies antialiased, floating-point advancements rendering mode (with printerFont). Available in Mac OS X v10.4 and later.
NSFontDefaultRenderingMode	0	One of the constants specify the font rendering mode. Determines the actual mode based on the user preference. Available in Mac OS X v10.4 and later.
NSFontIntegerAdvancementsRenderingMode	2	One of the constants specify the font rendering mode. Specifies integer advancements rendering mode. Available in Mac OS X v10.4 and later.
NSNativeShortGlyphPacking	5	A constant for glyph packing. The native format for Mac OS X. Carbon stores all text data as Unicode. The text engine stores glyph IDs and places them in 1-, 2-, or 4-byte short arrays. To render text, you must convert the storage format. Quartz understands. The following constants describe the rendering engine can use. They are used to create an array for making a multibyte (or single-byte) array of glyphs. For example, the window server, which expects glyphs in a short array (that is, "packed glyphs") instead of a pure NSGlyphGenerator glyphPacking. With Quartz, the engine always expects a short array, so NSNativeShortGlyphPacking is in use.
NSNullGlyph	0	One of the constants define reserved glyph codes. A null glyph.

4.12 class NSHelpManagerMBS

4.12.1 class NSHelpManagerMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The NSHelpManager class provides an approach to displaying online help.

Notes: An application contains one NSHelpManager object.

Blog Entries

- [MBS Xojo Plugins, version 20.5pr8](#)
- [MBS Xojo Plugins, version 19.3pr2](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS REALbasic Plugins, version 11.1pr5](#)

4.12.2 Methods

4.12.3 Constructor

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Creates an NSHelpManagerMBS object in Xojo which points to the shared NSHelpManager object.

4.12.4 findString(query as string, book as string)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Performs a search for the specified string in the specified book.

Notes: query: String to search for.

book: Localized help book to search. When "", all installed help books are searched.

4.12.5 isContextHelpModeActive as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Indicates whether context-sensitive help mode is active.

Notes: Returns true when the application is in context-sensitive help mode, false otherwise.

In context-sensitive help mode, when a user clicks a user interface item, help for that item is displayed in a small window just below the cursor.

4.12.6 NSContextHelpModeDidActivateNotification as string

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to be used with NSNotificationObserverMBS class.

Notes: Posted when the application enters context-sensitive help mode. This typically happens when the user holds down the Help key.

The notification object is the help manager. This notification does not contain a userInfo dictionary.

4.12.7 NSContextHelpModeDidDeactivateNotification as string

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to be used with NSNotificationObserverMBS class.

Notes: Posted when the application exits context-sensitive help mode. This happens when the user clicks the mouse button while the cursor is anywhere on the screen after displaying a context-sensitive help topic.

The notification object is the help manager. This notification does not contain a userInfo dictionary.

4.12.8 openHelpAnchor(anchor as string, book as string)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Finds and displays the text at the given anchor location in the given book.

Notes: anchor: Location of the desired text.

book: Help book containing the anchor. When "", all installed help books are searched.

4.12.9 registerBooksInBundle(bundle as NSBundleMBS) as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Registers one or more help books in the given bundle.

Notes: bundle: The bundle for additional help books. Books in the main bundle are automatically registered.

Returns true if registration is successful, false if the bundle doesn't contain any help books or if registration fails.

You use `registerBooksInBundle` to register help books in, for example, a plug-in bundle. The `Info.plist` in the bundle should contain a help book directory path, which specifies one or more folders containing help books.

The main bundle is automatically registered by `openHelpAnchor` and `findString`.

4.12.10 `setContextHelpModeActive(active as boolean)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Specifies whether context-sensitive help mode is active.

Notes: `active`: True turns on context-sensitive help, false turns it off.

You never send this message directly; instead, the `NSApplication` method `activateContextHelpMode` activates context-sensitive help mode, and the first mouse click after displaying the context-sensitive help window deactivates it.

When the application enters context-sensitive help mode, the help manager posts an `NSContextHelpModeDidActivateNotification` to the default notification center. When the application returns to normal operation, the help manager posts an `NSContextHelpModeDidDeactivateNotification`.

4.12.11 Properties

4.12.12 Handle as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The internal object reference to the `NSHelpManager` object.

Notes: (Read and Write property)

4.13 class NSIndexSetMBS

4.13.1 class NSIndexSetMBS

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: The NSIndexSet class represents an immutable collection of unique unsigned integers, known as indexes because of the way they are used.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

Notes: This collection is referred to as a index set.

You use index sets in your code to store indexes into some other data structure. For example, given an array, you could use an index set to identify a subset of objects in that array.

Each index value can appear only once in the index set. This is an important concept to understand and is why you would not use index sets to store an arbitrary collection of integer values. To illustrate how this works, if you created an NSIndexSet object with the values 4, 5, 2, and 5, the resulting set would only have the values 4, 5, and 2 in it. Because index values are always maintained in sorted order, the actual order of the values when you created the set would be 2, 4, and then 5.

In most cases, using an index set is more efficient than storing a collection of individual integers. Internally, the NSIndexSet class represents indexes using ranges. For maximum performance and efficiency, overlapping ranges in an index set are automatically coalesced—that is, ranges merge rather than overlap. Thus, the more contiguous the indexes in the set, the fewer ranges are required to specify those indexes.

You must not subclass the NSIndexSet class.

The mutable subclass of NSIndexSet is NSMutableIndexSet.

Blog Entries

- [MBS Real Studio Plugins, version 13.0pr7](#)

4.13.2 Methods

4.13.3 Constructor

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated `NSIndexSet` object.

Example:

```
dim x as new NSIndexSetMBS
```

See also:

- 4.13.4 Constructor(index as Integer) 286
- 4.13.5 Constructor(indexes as NSIndexSetMBS) 286
- 4.13.6 Constructor(StartIndex as Integer, Length as Integer) 287

4.13.4 Constructor(index as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated `NSIndexSet` object with an index.

Example:

```
dim n as new NSIndexSetMBS(5)
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

See also:

- 4.13.3 Constructor 285
- 4.13.5 Constructor(indexes as NSIndexSetMBS) 286
- 4.13.6 Constructor(StartIndex as Integer, Length as Integer) 287

4.13.5 Constructor(indexes as NSIndexSetMBS)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated `NSIndexSet` object with an index set.

Example:

```
dim n as new NSIndexSetMBS(5)
dim x as new NSIndexSetMBS(n)
MsgBox str(x.firstIndex)
```

See also:

4.13. CLASS NSINDEXSETMBS	287
• 4.13.3 Constructor	285
• 4.13.4 Constructor(index as Integer)	286
• 4.13.6 Constructor(StartIndex as Integer, Length as Integer)	287

4.13.6 Constructor(StartIndex as Integer, Length as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated NSMutableIndexSet object with an index range.

Example:

```
dim n as new NSMutableIndexSet(5,6) // 5, 6, 7, 8, 9, 10
```

Notes: This method raises an NSRangeException when indexRange would add an index that exceeds the maximum allowed value for unsigned integers.

See also:

• 4.13.3 Constructor	285
• 4.13.4 Constructor(index as Integer)	286
• 4.13.5 Constructor(indexes as NSMutableIndexSet)	286

4.13.7 containsIndex(index as Integer) as boolean

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Indicates whether the receiver contains a specific index.

Example:

```
dim n as new NSMutableIndexSet(5,6) // 5, 6, 7, 8, 9, 10
```

```
if n.containsIndex(6) then
  MsgBox "OK"
else
  MsgBox "Error."
end if
```

```
if n.containsIndex(11) then
  MsgBox "Error."
else
  MsgBox "OK"
end if
```

Notes: Returns true when the receiver contains index, false otherwise.

4.13.8 containsIndexes(indexes as NSIndexSetMBS) as boolean

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Indicates whether the receiver contains a superset of the indexes in another index set.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
dim m as new NSIndexSetMBS(6,2)

if n.containsIndexes(m) then
  MsgBox "OK"
else
  MsgBox "Error."
end if
```

Notes: True when the receiver contains a superset of the indexes in indexSet, false otherwise.

4.13.9 containsIndexesInRange(StartIndex as Integer, Length as Integer) as boolean

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Indicates whether the receiver contains the indexes represented by an index range.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
if n.containsIndexesInRange(6,2) then
  MsgBox "OK"
else
  MsgBox "Error."
end if
```

Notes: Returns true when the receiver contains the indexes in indexRange, false otherwise.

4.13.10 copy as NSIndexSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a copy of the Cocoa and RB object.

4.13.11 count as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns the number of indexes in the receiver.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.count) // shows 6
```

4.13.12 countOfIndexesInRange(StartIndex as Integer, Length as Integer) as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns the number of indexes in the receiver that are members of a given range.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.countOfIndexesInRange(1,8)) // shows 4
```

Notes: Available in Mac OS X v10.5 and later.

4.13.13 firstIndex as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the first index in the receiver or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

Notes: First index in the receiver or NSNotFound (&h7fffffff) when the receiver is empty.

4.13.14 `indexGreaterThanIndex(index as Integer) as Integer`

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the closest index in the receiver that is greater than a specific index or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.indexGreaterThanIndex(1)) // shows(5)
```

Notes: Returns the locest index in the receiver greater than index; `NSNotFound (&h7FFFFFFF)` when the receiver contains no qualifying index.

4.13.15 `indexGreaterThanOrEqualToIndex(index as Integer) as Integer`

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the closest index in the receiver that is greater than or equal to a specific index or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.indexGreaterThanOrEqualToIndex(1)) // shows(5)
```

Notes: Returns closest index in the receiver greater than or equal to index; `NSNotFound (&h7FFFFFFF)` when the receiver contains no qualifying index.

4.13.16 `indexLessThanIndex(index as Integer) as Integer`

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the closest index in the receiver that is less than a specific index or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.indexLessThanIndex(20)) // shows 10
MsgBox str(n.indexLessThanIndex(1)) // shows 2147483647 for not found
```

Notes: Returns closest index in the receiver less than index; `NSNotFound (&h7FFFFFFF)` when the re-

ceiver contains no qualifying index.

4.13.17 indexLessThanOrEqualToIndex(index as Integer) as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the closest index in the receiver that is less than or equal to a specific index or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.indexLessThanOrEqualToIndex(20)) // shows 10
```

Notes: Returns closest index in the receiver less than or equal to index; NSNotFound (&h7FFFFFFF) when the receiver contains no qualifying index.

4.13.18 indexSet as NSIndexSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an empty index set.

Example:

```
dim n as NSIndexSetMBS = NSIndexSetMBS.indexSet
MsgBox str(n.count) // 0 ->empty
```

Notes: Available in Mac OS X v10.3 and later.

4.13.19 indexSetWithIndex(index as Integer) as NSIndexSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an index set with an index.

Example:

```
dim n as NSIndexSetMBS = NSIndexSetMBS.indexSetWithIndex(3)
MsgBox str(n.count) // 1
```

Notes: Available in Mac OS X v10.3 and later.

4.13.20 `indexSetWithIndexesInRange(StartIndex as Integer, Length as Integer) as NSIndexSetMBS`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an index set with an index range.

Example:

```
dim n as NSIndexSetMBS = NSIndexSetMBS.indexSetWithIndexesInRange(3,2)
MsgBox str(n.count) // 2 and contains 3,4
```

Notes: Available in Mac OS X v10.3 and later.

4.13.21 `intersectsIndexesInRange(StartIndex as Integer, Length as Integer) as boolean`

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Indicates whether the receiver contains any of the indexes in a range.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
```

```
if n.intersectsIndexesInRange(1,4) then
MsgBox "Error"
else
MsgBox "OK"
end if
```

```
if n.intersectsIndexesInRange(1,6) then
MsgBox "OK"
else
MsgBox "Error"
end if
```

4.13.22 `isEqualToIndexSet(other as NSIndexSetMBS) as boolean`

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Indicates whether the indexes in the receiver are the same indexes contained in another index

set.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
dim m as NSIndexSetMBS = n.mutableCopy

if m.isEqualToIndexSet(n) then
  MsgBox "OK"
else
  MsgBox "Failed."
end if
```

4.13.23 lastIndex as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns either the last index in the receiver or the not-found indicator.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

Notes: Returns Last index in the receiver or NSNotFound (&h7FFFFFFF) when the receiver is empty.

4.13.24 mutableCopy as NSMutableIndexSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an editable copy of the indexset.

Example:

```
dim n as new NSIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
dim m as NSMutableIndexSetMBS = n.mutableCopy

m.addIndex 20

MsgBox str(n.lastIndex)+" "+str(m.lastIndex)
```

4.13.25 Operator_Convert as string

Plugin Version: 13.0, Platform: macOS, Targets: All.

Function: Converts an indexset to string for display.

Example:

```
dim n as NSIndexSetMBS = NSIndexSetMBS.indexSetWithIndexesInRange(10,40)
MsgBox n
```

Notes: This is for having str() function and msgbox work with NSIndexSetMBS class.
If more than 20 values, you get only 20 values followed with dots and last value on the end.

4.13.26 Values as Integer()

Plugin Version: 13.0, Platform: macOS, Targets: All.

Function: Returns all values in array.

Example:

```
dim n1 as NSIndexSetMBS = NSIndexSetMBS.indexSetWithIndexesInRange(10,10)
dim n2 as NSIndexSetMBS = NSIndexSetMBS.indexSetWithIndexesInRange(30,5)
dim n3 as new NSMutableIndexSetMBS
n3.addIndexes n1
n3.addIndexes n2
dim count1 as Integer = n1.count // 10
dim count2 as Integer = n2.count // 5
dim count3 as Integer = n3.count // 15
dim values1() as Integer = n1.Values
dim values2() as Integer = n2.Values
dim values3() as Integer = n3.Values
break // look in debugger
```

4.13.27 Properties

4.13.28 Handle as Integer

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: The internal reference for the NSIndexSex.

Notes: (Read and Write property)

4.14 class `NSInputStreamMBS`

4.14.1 class `NSInputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: A stream that provides read-only stream functionality.

Notes: Subclass of the `NSStreamMBS` class.

Blog Entries

- [MBS Xojo Plugins, version 18.2pr2](#)

4.14.2 Methods

4.14.3 `Constructor(filePath as string)`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSInputStream` object that reads data from the file at a given path.

Notes: The stream must be opened before it can be used.

4.14.4 `inputStreamWithData(data as Memoryblock) as NSInputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized `NSInputStream` object for reading from a given `NSData` object.

Notes: `data`: The data object from which to read. The contents of `data` are copied.

Returns an initialized `NSInputStream` object for reading from `data`. If `data` is not an `NSData` object, this method returns `nil`.

The stream must be opened before it can be used.

4.14.5 `inputStreamWithFilePath(path as string) as NSInputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSInputStream` object that reads data from the file at a given path.

Notes: `path`: The path to the file.

Returns an initialized `NSInputStream` object that reads data from the file at `path`.

The stream must be opened before it can be used.

4.14.6 inputStreamWithURL(URL as string) as NSInputStreamMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized NSInputStream object that reads data from the file at a given URL.

Notes: url: The URL to the file.

Returns an initialized NSInputStream object that reads data from the URL at url.

The stream must be opened before it can be used.

4.14.7 LookAhead as MemoryBlock

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Returns copy of read buffer.

Notes: Allows looking into data before reading it.

4.14.8 Read(maxLength as Integer) as MemoryBlock

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Reads up to a given number of bytes into a given buffer.

4.14.9 Properties

4.14.10 hasBytesAvailable as Boolean

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver has bytes available to read.

Notes: True if the receiver has bytes available to read, otherwise false. May also return true if a read must be attempted in order to determine the availability of bytes.

(Read only property)

4.15 class NSKeyedArchiverMBS

4.15.1 class NSKeyedArchiverMBS

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

Function: The plugin class for archiving objects.

Example:

```
// make archiver
dim a as new NSKeyedArchiverMBS

// add a string
a.encodeString "Hello World", "Greeting"

// finish
a.finishEncoding

// query data
dim m as MemoryBlock = a.archiverData

// start unarchiver
dim u as new NSKeyedUnarchiverMBS(m)

// query and display a value
dim s as string = u.decodeString("Greeting")
MsgBox s
```

Notes: NSKeyedArchiver, a concrete subclass of NSCoder, provides a way to encode objects (and scalar values) into an architecture-independent format that can be stored in a file. When you archive a set of objects, the class information and instance variables for each object are written to the archive. NSKeyedArchiver's companion class, NSKeyedUnarchiver, decodes the data in an archive and creates a set of objects equivalent to the original set.

A keyed archive differs from a non-keyed archive in that all the objects and values encoded into the archive are given names, or keys. When decoding a non-keyed archive, values have to be decoded in the same order in which they were encoded. When decoding a keyed archive, because values are requested by name, values can be decoded out of sequence or not at all. Keyed archives, therefore, provide better support for forward and backward compatibility.

The keys given to encoded values must be unique only within the scope of the current object being encoded. A keyed archive is hierarchical, so the keys used by object A to encode its instance variables do not conflict with the keys used by object B, even if A and B are instances of the same class. Within a single object, however, the keys used by a subclass can conflict with keys used in its superclasses.
Subclass of the NSCoderMBS class.

Blog Entries

- [MBS Xojo Plugins, version 21.2pr7](#)
- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)

4.15.2 Methods**4.15.3 archiverData as memoryblock**

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

Function: Returns the archived data.

4.15.4 Constructor

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

Function: Returns the receiver, initialized for encoding an archive.

See also:

- [4.15.5 Constructor\(RequiringSecureCoding as Boolean\)](#)

298

4.15.5 Constructor(RequiringSecureCoding as Boolean)

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

Function: Creates an archiver to encode data, and optionally disables secure coding.

Example:

```
Dim ckRecord As New CKRecordMBS("MyType")

// archive
Dim coder As NSKeyedArchiverMBS = New NSKeyedArchiverMBS(True)
ckRecord.encodeSystemFieldsWithCoder(coder)
coder.finishEncoding()

Dim encodedRecordValue As String = coder.archiverData

// unarchive
Dim unarchiver As New NSKeyedUnarchiverMBS(encodedRecordValue)
Dim decodedRecord As CKRecordMBS = New CKRecordMBS(unarchiver)

MsgBox "OK " + decodedRecord.recordType
```

[Exception](#) n As NSExcptionMBS
 MsgBox n.message

Notes: requiresSecureCoding: A Boolean value indicating whether all encoded objects must conform to NSSecureCoding.

To prevent the possibility of encoding an object that NSKeyedUnarchiver can,Ãt decode, set requiresSecureCoding to true whenever possible. This ensures that all encoded objects conform to NSSecureCoding.

Requires macOS 10.13.

See also:

- 4.15.4 Constructor

298

4.15.6 finishEncoding

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

Function: Instructs the receiver to construct the final data stream.

Notes: No more values can be encoded after this method is called. You must call this method when finished.

4.15.7 Properties

4.15.8 outputFormat as Integer

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

Function: Sets or queries the format in which the receiver encodes its data.

Example:

```
// make archiver
dim a as new NSKeyedArchiverMBS
a.outputFormat = a.kCFPropertyListXMLFormat_v1_0

// add a string
a.encodeString "Hello World", "Greeting"

// finish
a.finishEncoding

// query data and show
dim m as MemoryBlock = a.archiverData
dim s as string = DefineEncoding(m, encodings.UTF8)
```

MsgBox s

Notes: (Read and Write computed property)

4.15.9 Constants

Constants

Constant	Value	Description
kCFPropertyListBinaryFormat_v1_0	200	One of the possible formats. Specifies the binary property list format.
kCFPropertyListXMLFormat_v1_0	100	One of the possible formats. Specifies the XML property list format.

4.16 class NSKeyedUnarchiverMBS

4.16.1 class NSKeyedUnarchiverMBS

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

Function: The plugin class for unarchiving objects.

Example:

```

// make archiver
dim a as new NSKeyedArchiverMBS

// add a string
a.encodeString "Hello World", "Greeting"

// finish
a.finishEncoding

// query data
dim m as MemoryBlock = a.archiverData

// start unarchiver
dim u as new NSKeyedUnarchiverMBS(m)

// query and display a value
dim s as string = u.decodeString("Greeting")
MsgBox s

```

Notes: NSKeyedUnarchiver, a concrete subclass of NSCoder, defines methods for decoding a set of named objects (and scalar values) from a keyed archive. Such archives are produced by instances of the NSKeyedArchiver class.

A keyed archive is encoded as a hierarchy of objects. Each object in the hierarchy serves as a namespace into which other objects are encoded. The objects available for decoding are restricted to those that were encoded within the immediate scope of a particular object. Objects encoded elsewhere in the hierarchy, whether higher than, lower than, or parallel to this particular object, are not accessible. In this way, the keys used by a particular object to encode its instance variables need to be unique only within the scope of that object.

If you invoke one of the decode... methods of this class using a key that does not exist in the archive, a non-positive value is returned. This value varies by decoded type. For example, if a key does not exist in an archive, decodeBoolForKey returns false, decodeIntForKey returns 0, and decodeObjectForKey returns nil.

NSKeyedUnarchiver supports limited type coercion. A value encoded as any type of integer, whether a

standard int or an explicit 32-bit or 64-bit integer, can be decoded using any of the integer decode methods. Likewise, a value encoded as a float or double can be decoded as either a float or a double value. If an encoded value is too large to fit within the coerced type, the decoding method raises an NSRangeException. Further, when trying to coerce a value to an incompatible type, for example decoding an int as a float, the decoding method raises an NSInvalidUnarchiveOperationException.

Subclass of the NSCoderMBS class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)

4.16.2 Methods

4.16.3 Constructor(data as memoryblock)

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

Function: Initializes the receiver for decoding an archive previously encoded by NSKeyedArchiverMBS.

Notes: When you finish decoding data, you should invoke finishDecoding.

This method raises an NSExceptionMBS if data is not a valid archive.

4.16.4 finishDecoding

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

Function: Tells the receiver that you are finished decoding objects.

Notes: Invoking this method allows the receiver to notify its delegate and to perform any final operations on the archive. Once this method is invoked, the receiver cannot decode any further values.

4.17 class NSKeyValueObserverMBS

4.17.1 class NSKeyValueObserverMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The Plugin class to use a key value observer.

Notes: The NSKeyValueObserving (KVO) informal protocol defines a mechanism that allows objects to be notified of changes to the specified properties of other objects.

You can observe any object properties including simple attributes, to-one relationships, and to-many relationships. Observers of to-many relationships are informed of the type of change made —as well as which objects are involved in the change.

NSObject provides an implementation of the NSKeyValueObserving protocol that provides an automatic observing capability for all objects. You can further refine notifications by disabling automatic observer notifications and implementing manual notifications using the methods in this protocol.

This class implements an observer with event for use in Xojo.

Blog Entries

- [MBS Real Studio Plugins, version 13.1pr13](#)

4.17.2 Methods

4.17.3 addObserver(keyPath as string, options as Integer = 5, context as Variant = nil)

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Registers observer to receive KVO notifications for the specified key-path relative to the receiver.

Notes: keyPath: The key path, relative to the receiver, of the property to observe. This value must not be nil.

options: A combination of the option constants that specifies what is included in observation notifications.

context: Arbitrary data that is passed to anObserver in observeValueForKeyPath.

This class holds a strong references to context and a weak reference to target.

Available in OS X v10.3 and later.

4.17.4 Constructor(TargetHandle as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The constructor.

Notes: Please pass the handle of the target Cocoa object. Like `NSWindowMBS.handle` or `CALayerMBS`.

4.17.5 Destructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The destructor.

4.17.6 `NSKeyValueChangeIndexesKey` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for the dictionary passed to `observedValueForKeyPathChanged` event.

Notes: If the value of the `NSKeyValueChangeKindKey` entry is `NSKeyValueChangeInsertion`, `NSKeyValueChangeRemoval`, or `NSKeyValueChangeReplacement`, the value of this key is an `NSIndexSetMBS` object that contains the indexes of the inserted, removed, or replaced objects.

Available in OS X v10.3 and later.

4.17.7 `NSKeyValueChangeKindKey` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for the dictionary passed to `observedValueForKeyPathChanged` event.

4.17.8 `NSKeyValueChangeNewKey` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for the dictionary passed to `observedValueForKeyPathChanged` event.

Notes: If the value of the `NSKeyValueChangeKindKey` entry is `NSKeyValueChangeSetting`, and `NSKeyValueObservingOptionNew` was specified when the observer was registered, the value of this key is the new value for the attribute.

For `NSKeyValueChangeInsertion` or `NSKeyValueChangeReplacement`, if `NSKeyValueObservingOptionNew` was specified when the observer was registered, the value for this key is an `NSArray` instance that contains the objects that have been inserted or replaced other objects, respectively.

Available in OS X v10.3 and later.

4.17.9 NSKeyValueChangeNotificationIsPriorKey as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for the dictionary passed to `observedValueForKeyPathChanged` event.

Notes: If the option `NSKeyValueObservingOptionPrior` was specified when the observer was registered this notification is sent prior to a change.

The change dictionary contains an `NSKeyValueChangeNotificationIsPriorKey` entry whose value is a boolean. Available in OS X v10.5 and later.

4.17.10 NSKeyValueChangeOldKey as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for the dictionary passed to `observedValueForKeyPathChanged` event.

Notes: If the value of the `NSKeyValueChangeKindKey` entry is `NSKeyValueChangeSetting`, and `NSKeyValueObservingOptionOld` was specified when the observer was registered, the value of this key is the value before the attribute was changed.

For `NSKeyValueChangeRemoval` or `NSKeyValueChangeReplacement`, if `NSKeyValueObservingOptionOld` was specified when the observer was registered, the value is an `NSArray` instance that contains the objects that have been removed or have been replaced by other objects, respectively.

Available in OS X v10.3 and later.

4.17.11 `removeObserver(keyPath as string, context as Variant = nil)`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Stops a given object from receiving change notifications for the property specified by a given key-path relative to the receiver and a context.

Notes: `keyPath`: A key-path, relative to the receiver, for which observer is registered to receive KVO change notifications.

`context`: Arbitrary data that more specifically identifies the observer to be removed.

Examining the value in `context` you are able to determine precisely which `addObserver` method was used to create the observation relationship. When the same observer is registered for the same key-path multiple times, but with different context pointers, an application can determine specifically which object to stop observing.

The context object is only used in OS X v10.7 and later.

4.17.12 Properties

4.17.13 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.17.14 Events

4.17.15 `observedValueForKeyPathChanged(keyPath as string, target as Variant, change as dictionary, context as Variant, ChangeNSDictionaryRef as Integer)` as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: This event is called when the value at the specified key path relative to the given object has changed.

Notes: `keyPath`: The key path, relative to object, to the value that has changed.

`object`: The source object of the key path `keyPath`.

`change`: A dictionary that describes the changes that have been made to the value of the property at the key path `keyPath` relative to object. Entries are described in `Keys` used by the change dictionary.

`context`: The value that was provided when the receiver was registered to receive key-value observation notifications.

The receiver must be registered as an observer for the specified `keyPath` and object.

`ChangeNSDictionaryRef` provides for debugging the reference to the original `NSDictionary` object.

4.17.16 Constants

Change Kind Constants.

Constant	Value	Description
kChangeInsertion	2	Indicates that an object has been inserted into the to-many relationship that is being observed.
kChangeRemoval	3	Indicates that an object has been removed from the to-many relationship that is being observed.
kChangeReplacement	4	Indicates that an object has been replaced in the to-many relationship that is being observed.
kChangeSetting	1	Indicates that the value of the observed key path was set to a new value. This change can occur when observing an attribute of an object, as well as properties that specify to-one and to-many relationships.

Option Constants

Constant	Value	Description
kOptionInitial	4	<p>If specified, a notification should be sent to the observer immediately, before the observer registration method even returns.</p> <p>The change dictionary in the notification will always contain an <code>NSKeyValueChangeNewKey</code> entry if <code>kOptionNew</code> is also specified but will never contain an <code>NSKeyValueChangeOldKey</code> entry. (In an initial notification the current value of the observed property may be old, but it's new to the observer.) You can use this option instead of explicitly invoking, at the same time, code that is also invoked by the observer's <code>observedValueForKeyPathChanged</code> method. When this option is used with <code>addObserver</code> a notification will be sent for each indexed object to which the observer is being added.</p> <p>Available in OS X v10.5 and later.</p>
kOptionNew	1	<p>Indicates that the change dictionary should provide the new attribute value, if applicable.</p> <p>Available in OS X v10.3 and later.</p>
kOptionOld	2	<p>Indicates that the change dictionary should contain the old attribute value, if applicable.</p> <p>Available in OS X v10.3 and later.</p>
kOptionPrior	8	<p>Whether separate notifications should be sent to the observer before and after each change, instead of a single notification after the change.</p> <p>The change dictionary in a notification sent before a change always contains an <code>NSKeyValueChangeNotificationIsPriorKey</code> entry whose value is true, but never contains an <code>NSKeyValueChangeNewKey</code> entry. When this option is specified the change dictionary in a notification sent after a change contains the same entries that it would contain if this option were not specified. You can use this option when the observer's own key-value observing-compliance requires it to invoke one of the <code>-willChange...</code> methods for one of its own properties, and the value of that property depends on the value of the observed object's property. (In that situation it's too late to easily invoke <code>-willChange...</code> properly in response to receiving an <code>observedValueForKeyPathChanged</code> message after the change.)</p> <p>Available in OS X v10.5 and later.</p>

4.18 class NSMutableCharacterSetMBS

4.18.1 class NSMutableCharacterSetMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The NSMutableCharacterSet class declares the programmatic interface to objects that manage a modifiable set of Unicode characters.

Notes: You can add or remove characters from a mutable character set as numeric values in NSRange structures or as character values in strings, combine character sets by union or intersection, and invert a character set.

Mutable character sets are less efficient to use than immutable character sets. If you don't need to change a character set after creating it, create an immutable copy with `copy` and use that.

NSMutableCharacterSet defines no primitive methods. Subclasses must implement all methods declared by this class in addition to the primitives of NSMutableCharacterSet. They must also implement `mutableCopy`.

Subclass of the NSMutableCharacterSetMBS class.

Blog Entries

- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

- [7.6, page 8: News](#)

4.18.2 Methods

4.18.3 addCharactersInRange(aRange as NSRangeMBS)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Adds to the receiver the characters whose Unicode values are in a given range.

Example:

```
dim m as new NSMutableCharacterSetMBS
dim r as new NSRangeMBS(65,3)
m.addCharactersInRange r
MsgBox m // shows "ABC"
```

Notes: `aRange`: The range of characters to add.

`aRange.location` is the value of the first character to add; `aRange.location + aRange.length-1` is the value

of the last. If `aRange.length` is 0, this method has no effect.

4.18.4 `addCharactersInString(aString as string)`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Adds to the receiver the characters in a given string.

Example:

```
dim m as new NSMutableCharacterSetMBS
m.addCharactersInString "Hello"
MsgBox m // shows "Helo"
```

Notes: This method has no effect if `aString` is empty.

4.18.5 Constructor

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an empty mutable character set.

4.18.6 `formIntersectionWithCharacterSet(otherset as NSMutableCharacterSetMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Modifies the receiver so it contains only characters that exist in both the receiver and `otherSet`.

4.18.7 `formUnionWithCharacterSet(otherset as NSMutableCharacterSetMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Modifies the receiver so it contains all characters that exist in either the receiver or `otherSet`.

4.18.8 `invert`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Replaces all the characters in the receiver with all the characters it didn't previously contain.

Notes: Inverting a mutable character set, whether by `invert` or by `invertedSet`, is much less efficient than

inverting an immutable character set with `invertedSet`.

4.18.9 `removeCharactersInRange(aRange as NSRangeMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Removes from the receiver the characters whose Unicode values are in a given range.

Example:

```
dim m as new NSMutableCharacterSetMBS
dim r as new NSRangeMBS(65,3)
m.addCharactersInRange new NSRangeMBS(65,6)
m.removeCharactersInRange r
MsgBox m // shows "DEF"
```

Notes: `aRange`: The range of characters to remove.

`aRange.location` is the value of the first character to remove; `aRange.location + aRange.length-1` is the value of the last. If `aRange.length` is 0, this method has no effect.

4.18.10 `removeCharactersInString(aString as string)`

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Removes from the receiver the characters in a given string.

Example:

```
dim m as new NSMutableCharacterSetMBS
m.addCharactersInString "Hello"
m.removeCharactersInString "World"
MsgBox m // shows "He"
```

Notes: This method has no effect if `aString` is empty.

4.19 class NSMutableIndexSetMBS

4.19.1 class NSMutableIndexSetMBS

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: The NSMutableIndexSet class represents a mutable collection of unique unsigned integers, known as indexes because of the way they are used.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

Notes: This collection is referred to as a mutable index set.

The values in a mutable index set are always sorted, so the order in which values are added is irrelevant.

You must not subclass the NSMutableIndexSet class.
Subclass of the NSIndexSetMBS class.

4.19.2 Methods

4.19.3 addIndex(index as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Adds an index to the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
n.addIndex 12
MsgBox str(n.count)
```

4.19.4 addIndexes(indexes as NSIndexSetMBS)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Adds the indexes in an index set to the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
```

```
dim x as new NSIndexSetMBS(12,5)
n.addIndexes x
MsgBox str(n.count)
```

4.19.5 addIndexesInRange(StartIndex as Integer, Length as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Adds the indexes in an index range to the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
n.addIndexesInRange 12,5
MsgBox str(n.count)
```

Notes: Index range to add. Must include only indexes representable as unsigned integers.

This method raises an NSRangeException when indexRange would add an index that exceeds the maximum allowed value for unsigned integers.

4.19.6 Constructor

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated NSMutableIndexSet object.

Example:

```
dim x as new NSMutableIndexSetMBS
```

See also:

- 4.19.7 Constructor(index as Integer) 312
- 4.19.8 Constructor(indexes as NSIndexSetMBS) 313
- 4.19.9 Constructor(StartIndex as Integer, Length as Integer) 313

4.19.7 Constructor(index as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated NSMutableIndexSet object with an index.

Example:

```
dim n as new NSMutableIndexSetMBS(5)
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

See also:

- 4.19.6 Constructor 312
- 4.19.8 Constructor(indexes as NSIndexSetMBS) 313
- 4.19.9 Constructor(StartIndex as Integer, Length as Integer) 313

4.19.8 Constructor(indexes as NSIndexSetMBS)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated NSIndexSet object with an index set.

Example:

```
dim n as new NSMutableIndexSetMBS(5)
dim x as new NSMutableIndexSetMBS(n)
MsgBox str(x.firstIndex)
```

See also:

- 4.19.6 Constructor 312
- 4.19.7 Constructor(index as Integer) 312
- 4.19.9 Constructor(StartIndex as Integer, Length as Integer) 313

4.19.9 Constructor(StartIndex as Integer, Length as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Initializes an allocated NSIndexSet object with an index range.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
```

Notes: This method raises an NSRangeException when indexRange would add an index that exceeds the maximum allowed value for unsigned integers.

See also:

- 4.19.6 Constructor 312
- 4.19.7 Constructor(index as Integer) 312
- 4.19.8 Constructor(indexes as NSIndexSetMBS) 313

4.19.10 removeAllIndexes

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Removes the receiver's indexes.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
n.removeAllIndexes
MsgBox str(n.count)
```

4.19.11 removeIndex(index as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Removes an index from the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
MsgBox str(n.count)
n.removeIndex 8
MsgBox str(n.count)
```

4.19.12 removeIndexes(indexes as NSIndexSetMBS)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Removes the indexes in an index set from the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
dim x as new NSIndexSetMBS(5,2)
n.removeIndexes x
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

4.19.13 removeIndexesInRange(StartIndex as Integer, Length as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Removes the indexes in an index range from the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
n.removeIndexesInRange 5,2
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

4.19.14 shiftIndexes(StartingAtIndex as Integer, delta as Integer)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Shifts a group of indexes to the left or the right within the receiver.

Example:

```
dim n as new NSMutableIndexSetMBS(5,6) // 5, 6, 7, 8, 9, 10
n.shiftIndexes 7,3
MsgBox str(n.firstIndex)+" "+str(n.lastIndex)
```

Notes: startIndex: Head of the group of indexes to shift.

delta: Amount and direction of the shift. Positive integers shift the indexes to the right. Negative integers shift the indexes to the left.

The group of indexes shifted is made up by startIndex and the indexes that follow it in the receiver.

A left shift deletes the indexes in the range (startIndex-delta,delta) from the receiver.

A right shift inserts empty space in the range (indexStart,delta) in the receiver.

4.20 class NSMutableURLRequestMBS

4.20.1 class NSMutableURLRequestMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The Cocoa class for a mutable URL Request.

Notes: NSMutableURLRequest is a subclass of NSURLRequest provided to aid developers who may find it more convenient to mutate a single request object for a series of URL load requests instead of creating an immutable NSURLRequest for each load.

This programming model is supported by the following contract between NSMutableURLRequest and NSURLConnection: NSURLConnection makes a deep copy of each NSMutableURLRequest object passed to one of its initializers.

Subclass of the NSURLRequestMBS class.

Blog Entries

- [Notes from the last days](#)
- [MBS Real Studio Plugins, version 11.3pr6](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

- [7.6, page 8: News](#)

4.20.2 Methods

4.20.3 addValue(value as string, field as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Adds an HTTP header to the receiver's HTTP header dictionary.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
```

```
m.addValue "Hello", "test"
```

```
m.addValue "World", "test"
```

```
MsgBox m.valueForHTTPHeaderField("test")
```

Notes: value: The value for the header field.

field: The name of the header field. In keeping with the HTTP RFC, HTTP header field names are case-

insensitive

This method provides the ability to add values to header fields incrementally. If a value was previously set for the specified field, the supplied value is appended to the existing value using the appropriate field delimiter. In the case of HTTP, this is a comma.

4.20.4 Constructor(url as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
```

MsgBox m.URL

Notes: Default values are used for cache policy (NSURLRequestUseProtocolCachePolicy) and timeout interval (60 seconds).

On success, handle property is not zero.

See also:

- 4.20.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double) 317

4.20.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

See also:

- 4.20.4 Constructor(url as string) 317

4.20.6 setAllHTTPHeaderFields(headerFields as Dictionary)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Replaces the receiver's header fields with the passed values.

Notes: headerFields: A dictionary with the new header fields. HTTP header fields must be string values; therefore, each object and key in the headerFields dictionary must be a String. If either the key or value for a key-value pair is not a string, the key-value pair is skipped.

4.20.7 setAttribution(Attribution as Integer)

Plugin Version: 21.5, Platform: macOS, Targets: All.

Function: Sets the NSURLRequestAttribution to associate with this request.

Notes: Defaults to NSURLRequestAttributionDeveloper.

4.20.8 setCachePolicy(policy as Integer)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Sets the cache policy of the receiver.

Notes: policy: The new cache policy.

Available in Mac OS X v10.2 with Safari 1.0 installed.

Available in Mac OS X v10.2.7 and later.

These constants are used to specify interaction with the cached responses.

NSURLRequestUseProtocolCachePolicy = 0,

NSURLRequestReloadIgnoringLocalCacheData = 1,

NSURLRequestReloadIgnoringLocalAndRemoteCacheData = 4,

NSURLRequestReturnCacheDataElseLoad = 2,

NSURLRequestReturnCacheDataDontLoad = 3,

NSURLRequestReloadRevalidatingCacheData = 5

4.20.9 setHTTPBody(data as MemoryBlock)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets the request body of the receiver to the specified data.

Notes: data: The new request body for the receiver. This is sent as the message body of the request, as in an HTTP POST request.

4.20.10 setHTTPMethod(HTTPMethod as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets the receiver's HTTP request method.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setHTTPMethod "PUT"
```

MsgBox m.HTTPMethod

Notes: HTTPMethod: The new HTTP request method. The default HTTP method is "GET".

4.20.11 setHTTPSShouldHandleCookies(should as boolean)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets whether the receiver should use the default cookie handling for the request.

Example:

```
dim r as NSURLRequestMBS = NSURLRequestMBS.requestWithURL("http://www.apple.de")
Dim m as NSMutableURLRequestMBS = r.mutableCopy
m.setHTTPSShouldHandleCookies(false)
HTMLViewer1.LoadRequest m
```

Notes: handleCookies: True if the receiver should use the default cookie handling for the request, false otherwise. The default is true.

4.20.12 setHTTPSShouldUsePipelining(shouldUsePipelining as boolean)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets whether the request should not wait for the previous response before transmitting.

Notes: True if the receiver should transmit before the previous response is received. False to wait for the previous response before transmitting.

Calling this method with a true value does not guarantee HTTP pipelining behavior. HTTP 1.1 allows the client to send multiple requests to the server without waiting for a response.

Though HTTP 1.1 requires support for pipelining, some servers report themselves as being HTTP 1.1 but do not support pipelining. To maintain compatibility with these servers, requests may have to wait for the previous response before transmitting.

Available on Mac OS X 10.7 or newer.

4.20.13 setMainDocumentURL(url as string)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Sets the main document URL for the receiver.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test/test.jpg")
m.setMainDocumentURL "http://test.test/"
MsgBox m.mainDocumentURL
```

Notes: url: The new main document URL. Can be nil.

The caller should set the main document URL to an appropriate main document, if known. For example, when loading a web page the URL of the HTML document for the top-level frame would be appropriate. This URL will be used for the "only from same domain as main document" cookie accept policy. Available in Mac OS X v10.2 with Safari 1.0 installed. Available in Mac OS X v10.2.7 and later.

4.20.14 setNetworkServiceType(networkServiceType as Integer)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets the NSURLRequestNetworkServiceType to associate with this request.

Notes: networkServiceType: The service type to associate with the request.

This method is used to provide the network layers with a hint as to the purpose of the request. Most clients should not need to use this method. See NSURLNetworkServiceType* constants.

Available in Mac OS X 10.7 or newer.

4.20.15 setTimeoutInterval(seconds as Double)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Sets the receiver's timeout interval, in seconds.

Notes: seconds: The timeout interval, in seconds. If during a connection attempt the request remains idle for longer than the timeout interval, the request is considered to have timed out. The default timeout interval is 60 seconds.

Available in Mac OS X v10.2 with Safari 1.0 installed.

Available in Mac OS X v10.2.7 and later.

4.20.16 setURL(url as string)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Sets the URL of the receiver.

Notes: Available in Mac OS X v10.2 with Safari 1.0 installed.
Available in Mac OS X v10.2.7 and later.

4.20.17 setValue(value as string, field as string)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Sets the specified HTTP header field.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setValue("just a test", "test")
MsgBox m.valueForHTTPHeaderField("test")
```

Notes: value: The new value for the header field. Any existing value for the field is replaced by the new value.

field: The name of the header field to set. In keeping with the HTTP RFC, HTTP header field names are case-insensitive.

4.21 class `NSOutputStreamMBS`

4.21.1 class `NSOutputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: A stream that provides write-only stream functionality.

Notes: Subclass of the `NSSStreamMBS` class.

Blog Entries

- [MBS Xojo Plugins, version 18.2pr2](#)

4.21.2 Methods

4.21.3 Constructor

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized output stream that will write stream data to memory.

Notes: Returns an initialized output stream that will write stream data to memory.

The stream must be opened before it can be used.

You retrieve the contents of the memory stream by `OutputData` property.

See also:

- [4.21.4 Constructor\(filePath as string, append as boolean\)](#)

322

4.21.4 Constructor(filePath as string, append as boolean)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized output stream for writing to a specified file.

Notes: `path`: The path to the file the output stream will write to.

`append`: True if newly written data should be appended to any existing file contents, otherwise false.

Returns an initialized output stream that can write to `path`.

The stream must be opened before it can be used.

See also:

- [4.21.3 Constructor](#)

322

4.21.5 `OutputData` as `MemoryBlock`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The output of the memory stream.

4.21.6 `outputStreamToFileAtPath(filePath as string, append as boolean)` as `NSOutputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized output stream for writing to a specified file.

Notes: `path`: The path to the file the output stream will write to.

`append`: True if newly written data should be appended to any existing file contents, otherwise false.

Returns an initialized output stream that can write to `path`.

The stream must be opened before it can be used.

4.21.7 `outputStreamToMemory` as `NSOutputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized output stream that will write stream data to memory.

Notes: Returns an initialized output stream that will write stream data to memory.

The stream must be opened before it can be used.

You retrieve the contents of the memory stream by `OutputData` property.

4.21.8 `outputStreamWithURL(fileURL as string, append as boolean)` as `NSOutputStreamMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates and returns an initialized output stream for writing to a specified URL.

Notes: `fileURL`: The URL to the file the output stream will write to.

`append`: True if newly written data should be appended to any existing file contents, otherwise false.

Returns an initialized output stream that can write to url.

The stream must be opened before it can be used.

4.21.9 write(data as MemoryBlock) as Integer

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Writes the contents of a provided data buffer to the receiver.

Notes: Returns a number indicating the outcome of the operation:

A positive number indicates the number of bytes written.

0 indicates that a fixed-length stream and has reached its capacity.

-1 means that the operation failed; more information about the error can be obtained with streamError.

4.21.10 Properties

4.21.11 hasSpaceAvailable as Boolean

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: A boolean value that indicates whether the receiver can be written to.

Notes: True if the receiver can be written to or if a write must be attempted in order to determine if space is available, false otherwise.

(Read only property)

4.22 class NSPanelMBS

4.22.1 class NSPanelMBS

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: The NSPanel class implements a special kind of window (known as a panel), typically performing an auxiliary function.

Example:

```
static n as NSPanelMBS
```

```
dim style as Integer = NSPanelMBS.NSClosableWindowMask + NSPanelMBS.NSTitledWindowMask +
NSPanelMBS.NSUtilityWindowMask + NSPanelMBS.NSHUDWindowMask + NSPanelMBS.NSResizableWin-
dowMask
```

```
n = new NSPanelMBS(100,100,200,200, style, NSPanelMBS.NSWindowBackingLocationDefault, false)
```

```
n.Title = "Hello World"
```

```
n.Show
```

Notes: Subclass of the NSWindowMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr8](#)
- [HUD Windows in REALbasic](#)
- [MonkeyBread Software Releases the MBS Plugins 8.2](#)

4.22.2 Methods

4.22.3 Constructor(x as Double, y as Double, w as Double, h as Double, style-Mask as Integer, BackingStoreType as Integer, deferCreation as boolean)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new Cocoa Windows.

Example:

```
static n as NSPanelMBS
```

```
dim style as Integer = NSPanelMBS.NSClosableWindowMask + NSPanelMBS.NSTitledWindowMask +
NSPanelMBS.NSUtilityWindowMask + NSPanelMBS.NSHUDWindowMask + NSPanelMBS.NSResizableWin-
dowMask
```

```
n = new NSPanelMBS(100,100,200,200, style, NSPanelMBS.NSWindowBackingLocationDefault, false)
n.Title = "Hello World"
n.Show
```

Notes: x,y,w,h:

Location and size of the window's content area in screen coordinates. Note that the window server limits window position coordinates to $\mp 16,000$ and sizes to 10,000.

styleMask:

The window's style. Either it can be `NSBorderlessWindowMask`, or it can contain any of the options described in the constants, combined using the `bitwiseOR` function. Borderless windows display none of the usual peripheral elements and are generally useful only for display or caching purposes; you should normally not need to create them. Also, note that a window's style mask should include `NSTitledWindowMask` if it includes any of the others.

bufferingType:

Specifies how the drawing done in the window is buffered by the window device, and possible values are described in "Constants."

deferCreation:

Specifies whether the window server creates a window device for the window immediately. When true, the window server defers creating the window device until the window is moved onscreen. All display messages sent to the window or its views are postponed until the window is created, just before it's moved onscreen.

Initialized `NSWindow` object.

This method is the designated initializer for the `NSWindow` class.

Deferring the creation of the window improves launch time and minimizes the virtual memory load on the window server.

The new window creates a view to be its default content view. You can replace it with your own object by using the `ContentView` property.

4.22.4 `RunAlertPanel(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string)` as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Runs an alert panel.

4.22.5 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer

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

Function: Run an alert panel.

See also:

- 4.22.6 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 327
- 4.22.7 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 327

4.22.6 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Run an alert panel.

See also:

- 4.22.5 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 327
- 4.22.7 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 327

4.22.7 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Run an alert panel.

See also:

- 4.22.5 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 327
- 4.22.6 RunAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 327

4.22.8 `RunCriticalAlertPanel`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Runs a critical alert panel.

4.22.9 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer

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

Function: Runs a critical alert panel.

See also:

- 4.22.10 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 328
- 4.22.11 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 328

4.22.10 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Runs a critical alert panel.

See also:

- 4.22.9 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 328
- 4.22.11 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 328

4.22.11 `RunCriticalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Runs a critical alert panel.

See also:

- 4.22.9 RunCriticalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 328
- 4.22.10 RunCriticalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 328

4.22.12 RunInformationalAlertPanel(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Run an informational alert panel.

4.22.13 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer

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

Function: Run an informational alert panel.

See also:

- 4.22.14 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 329
- 4.22.15 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 330

4.22.14 RunInformationalAlertPanelRelativeToWindow(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Run an informational alert panel.

See also:

- 4.22.13 `RunInformationalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 329
- 4.22.15 `RunInformationalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer 330

4.22.15 `RunInformationalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as window) as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Run an informational alert panel.

See also:

- 4.22.13 `RunInformationalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as DesktopWindow) as integer 329
- 4.22.14 `RunInformationalAlertPanelRelativeToWindow`(title as string, message as string, defaultButton as string, alternateButton as string, otherButton as string, docWindow as NSWindowMBS) as Integer 329

4.22.16 Properties

4.22.17 `becomesKeyOnlyIfNeeded` as boolean

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: whether the panel becomes the key window only when needed.

Notes: By default, this attribute is set to false, indicating that the panel becomes key as other windows do. (Read and Write computed property)

4.22.18 `isFloatingPanel` as boolean

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Whether the panel is a floating panel.

Notes: True when the receiver is a floating panel, false otherwise.

By default, panels do not float above other windows. It's appropriate for an panel to float above other windows only if all of the following conditions are true:

- It is small enough not to obscure whatever is behind it.
- It is oriented more to the mouse than to the keyboard—that is, if it doesn't become the key window or becomes so only when needed.
- It needs to remain visible while the user works in the application's normal windows—for example, if the user must frequently move the cursor back and forth between a normal window and the panel (such as a tool palette), or if the panel gives information relevant to the user's actions in a normal window.
- It hides when the application is deactivated (the default behavior for panels).

(Read and Write computed property)

4.22.19 worksWhenModal as boolean

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver receives keyboard and mouse events even when some other window is being run modally.

Notes: True when the receiver receives keyboard and mouse events even when some other window is being run modally, false otherwise.

By default, this attribute is set to false, indicating a panel's ineligibility for events during a modal loop or session.

(Read and Write computed property)

4.22.20 Constants

Constants

Constant	Value	Description
NSAlertAlternateReturn	0	One of the possible return values for the alert panels. The user pressed the alternate button.
NSAlertDefaultReturn	1	One of the possible return values for the alert panels. The user pressed the default button.
NSAlertErrorReturn	-2	One of the possible return values for the alert panels. The alert cannot identify the reason it was closed; it may have been closed by an external source or by a button other than those listed above.
NSAlertOtherReturn	-1	One of the possible return values for the alert panels. The user pressed a second alternate button.
NSCancelButton	0	One of the constants for the modal panel return values. The Cancel button.
NSDocModalWindowMask	64	One of the constants you can use to specify the style when creating a window. The panel is created as a modal sheet.
NSHUDWindowMask	8192	One of the constants you can use to specify the style when creating a window.
NSNonactivatingPanelMask	128	One of the constants you can use to specify the style when creating a window. The panel can receive keyboard input without activating the owning application.
NSOKButton	1	One of the constants for the modal panel return values. The OK button.
NSUtilityWindowMask	16	One of the constants you can use to specify the style when creating a window. The panel is created as a floating window.

4.23 class NSResponderMBS

4.23.1 class NSResponderMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: NSResponder is an abstract class that forms the basis of event and command processing in the Application Kit.

Notes: See the Cocoa documentation for more details.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 13.2pr4](#)
- [MBS Real Studio Plugins, version 12.4pr8](#)
- [MBS REALbasic plug-in 9.6](#)

4.23.2 Methods

4.23.3 beginGestureWithEvent(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has begun a touch gesture.

Notes: The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

4.23.4 cancelOperation

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to cancel the current operation.

Notes: This method is bound to the Escape and Command-. (period) keys. The key window first searches the view hierarchy for a view whose key equivalent is Escape or Command-., whichever was entered. If none of these views handles the key equivalent, the window sends a default action message of cancelOperation: to the first responder and from there the message travels up the responder chain.

If no responder in the responder chain implements cancelOperation:, the key window searches the view hierarchy for a view whose key equivalent is Escape (note that this may be redundant if the original key equivalent was Escape). If no such responder is found, then a cancel: action message is sent to the first responder in the responder chain that implements it.

NSResponder declares but does not implement this method.

4.23.5 capitalizeWord

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to capitalize the word or words surrounding the insertion point or selection, expanding the selection if necessary.

Notes: If either end of the selection partially covers a word, that entire word is made lowercase. The sender argument is typically the object that invoked this method. `NSResponder` declares but doesn't implement this method.

4.23.6 centerSelectionInVisibleArea

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the selection, whatever it is, inside its visible area.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.7 changeCaseOfLetter

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to change the case of a letter or letters in the selection, perhaps by opening a panel with capitalization options or by cycling through possible case combinations.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.8 complete

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to complete an operation in progress or a partially constructed element.

Notes: This method can be interpreted, for example, as a request to attempt expansion of a partial word, such as for expanding a glossary shortcut, or to close a graphics item being drawn. `NSResponder` declares but doesn't implement this method.

4.23.9 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

4.23.10 cursorUpdate(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the mouse cursor has moved into a cursor rectangle.

Notes: Override this method to set the cursor image. The default implementation uses cursor rectangles, if cursor rectangles are currently valid. If they are not, it calls super to send the message up the responder chain.

If the responder implements this method, but decides not to handle a particular event, it should invoke the superclass implementation of this method.

Available in Mac OS X v10.5 and later.

4.23.11 deleteBackward

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or a single element backward from the insertion point (a letter or character in text, for example).

Notes: NSResponder declares but doesn't implement this method.

4.23.12 deleteBackwardByDecomposingPreviousCharacter

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or a single character backward from the insertion point.

Notes: If the previous character is canonically decomposable, this method should try to delete only the last character in the grapheme cluster (for example, deleting "a" + "̂" results in "a"). NSResponder declares but does not implement this method.

4.23.13 deleteForward

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or a single element forward from the insertion point (a letter or character in text, for example).

Notes: NSResponder declares but doesn't implement this method.

4.23.14 deleteToBeginningOfLine

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or all text from the insertion point to the beginning of a line (typically of text).

Notes: Also places the deleted text into the kill buffer. `NSResponder` declares but doesn't implement this method.

4.23.15 `deleteToBeginningOfParagraph`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or all text from the insertion point to the beginning of a paragraph of text.

Notes: Also places the deleted text into the kill buffer. `NSResponder` declares but doesn't implement this method.

4.23.16 `deleteToEndOfLine`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or all text from the insertion point to the end of a line (typically of text).

Notes: Also places the deleted text into the kill buffer. `NSResponder` declares but doesn't implement this method.

4.23.17 `deleteToEndOfParagraph`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or all text from the insertion point to the end of a paragraph of text.

Notes: Also places the deleted text into the kill buffer. `NSResponder` declares but doesn't implement this method.

4.23.18 `deleteToMark`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or all items from the insertion point to a previously placed mark, including the selection itself if not empty.

Notes: Also places the deleted text into the kill buffer. `NSResponder` declares but doesn't implement this method.

4.23.19 deleteWordBackward

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or a single word backward from the insertion point.

Notes: NSResponder declares but doesn't implement this method.

4.23.20 deleteWordForward

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to delete the selection, if there is one, or a single word forward from the insertion point.

Notes: NSResponder declares but doesn't implement this method.

4.23.21 endGestureWithEvent(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has ended a touch gesture.

Notes: Available in Mac OS X v10.6 and later.

4.23.22 flagsChanged(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has pressed or released a modifier key (Shift, Control, and so on).

Notes: The default implementation simply passes this message to the next responder.

4.23.23 flushBufferedKeyEvents

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Overridden by subclasses to clear any unprocessed key events.

4.23.24 helpRequested(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Displays context-sensitive help for the receiver if such exists; otherwise passes this message to the next responder.

Notes: `NSWindow` invokes this method automatically when the user clicks for help—while processing `theEvent`. Subclasses need not override this method, and application code shouldn't directly invoke it.

4.23.25 indent

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to indent the selection or the insertion point if there is no selection.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.26 insertBacktab

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to handle a backward tab.

Notes: A field editor might respond to this method by selecting the field before it, while a regular text object either doesn't respond to or ignores such a message. `NSResponder` declares but doesn't implement this method.

4.23.27 insertContainerBreak

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a container break (typically a page break) at the insertion point or selection, deleting the selection if there is one.

Notes: `NSResponder` declares but doesn't implement this method. `NSTextView` implements it to insert an `NSFormFeedCharacter` character (0x000c).

4.23.28 insertDoubleQuoteIgnoringSubstitution

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a double quote character at the insertion point without interference by automatic quote correction.

Notes: `NSResponder` declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.29 insertLineBreak

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a line break (as distinguished from a paragraph break) at the insertion point or selection, deleting the selection if there is one.

Notes: NSResponder declares but doesn't implement this method. NSTextView implements it to insert an NSLineSeparatorCharacter character (0x2028).

4.23.30 insertNewline

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a newline character at the insertion point or selection, deleting the selection if there is one, or to end editing if the receiver is a text field or other field editor.

Notes: NSResponder declares but doesn't implement this method.

4.23.31 insertNewlineIgnoringFieldEditor

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a line-break character at the insertion point or selection, deleting the selection if there is one.

Notes: Unlike insertNewline:, this method always inserts a line-break character and doesn't cause the receiver to end editing. NSResponder declares but doesn't implement this method.

4.23.32 insertParagraphSeparator

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a paragraph separator at the insertion point or selection, deleting the selection if there is one.

Notes: NSResponder declares but doesn't implement this method.

4.23.33 insertSingleQuoteIgnoringSubstitution

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a single quote character at the insertion point without interference by automatic quote correction.

Notes: NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.34 insertTab

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a tab character at the insertion point or selection, deleting the selection if there is one, or to end editing if the receiver is a text field or other field editor.

Notes: NSResponder declares but doesn't implement this method.

4.23.35 insertTabIgnoringFieldEditor

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to insert a tab character at the insertion point or selection, deleting the selection if there is one.

Notes: Unlike insertTab:, this method always inserts a tab character and doesn't cause the receiver to end editing. NSResponder declares but doesn't implement this method.

4.23.36 keyDown(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has pressed a key.

Notes: The receiver can interpret theEvent itself, or pass it to the system input manager using interpretKeyEvents. The default implementation simply passes this message to the next responder.

4.23.37 keyUp(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has released a key.

Notes: The default implementation simply passes this message to the next responder.

4.23.38 lowercaseWord

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to make lowercase every letter in the word or words surrounding the insertion point or selection, expanding the selection if necessary.

Notes: If either end of the selection partially covers a word, that entire word is made lowercase. `NSResponder` declares, but doesn't implement this method.

4.23.39 `magnifyWithEvent(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has begun a pinch gesture.

Notes: The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

4.23.40 `makeBaseWritingDirectionLeftToRight`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the paragraph base writing direction to be left to right.

Notes: Sets the `NSAttributedString` key `NSWritingDirectionAttributeName` to `NSWritingDirectionLeftToRight`.

This action method is intended to be used both as the target of a menu item and for key bindings. The base writing direction methods should be the target of three menu items in a submenu, under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.41 `makeBaseWritingDirectionNatural`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the paragraph base writing direction to be natural.

Notes: Natural directionality is determined from the text in accordance with the Unicode bi-di algorithm. For more information see `NSParagraphStyle`.

Sets the `NSAttributedString` key `NSWritingDirectionAttributeName` to `NSTextWritingDirectionEmbedding`.

This action method is intended to be used both as the target of a menu item and for key bindings. The base writing direction methods should be the target of three menu items in a submenu, under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.42 `makeBaseWritingDirectionRightToLeft`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the paragraph base writing direction to be right to left.

Notes: Sets the NSAttributedString key `NSWritingDirectionAttributeName` to `NSWritingDirectionRightToLeft`.

This action method is intended to be used both as the target of a menu item and for key bindings. The base writing direction methods should be the target of three menu items in a submenu, under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.43 `makeTextWritingDirectionLeftToRight`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the character level attributed string direction attribute for left to right text.

Notes: Sets the NSAttributedString `NSWritingDirectionAttributeName` to `NSWritingDirectionLeftToRight`.

This action method is intended to be used both as the target of a menu item and for key bindings. The text writing directions should be the target of three similar menu items in a submenu under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.44 `makeTextWritingDirectionNatural`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the character-level writing direction attribute.

Notes: Removes the `NSWritingDirectionAttributeName` from an `NSAttributedString`.

This action method is intended to be used both as the target of a menu item and for key bindings. The text writing directions should be the target of three similar menu items in a submenu under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.45 `makeTextWritingDirectionRightToLeft`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the character-level writing direction attribute to a single right-to-left embedding.

Notes: Sets the `NSAttributedString` key `NSWritingDirectionAttributeName` to `NSWritingDirectionRightToLeft`.

This action method is intended to be used both as the target of a menu item and for key bindings. The text writing directions should be the target of three similar menu items in a submenu under the Edit menu.

Default key bindings will also be provided for this method but will only be enabled for users of Hebrew or Arabic, or those who have otherwise enabled a suitable preference.

Available in Mac OS X v10.6 and later.

4.23.46 `mouseDown(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has pressed the left mouse button.

Notes: The default implementation simply passes this message to the next responder.

4.23.47 mouseDragged(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has moved the mouse with the left button pressed.

Notes: The default implementation simply passes this message to the next responder.

4.23.48 mouseEntered(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the cursor has entered a tracking rectangle.

Notes: The default implementation simply passes this message to the next responder.

Available in Mac OS X v10.0 and later.

4.23.49 mouseExited(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the cursor has exited a tracking rectangle.

Notes: The default implementation simply passes this message to the next responder.

4.23.50 mouseMoved(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the mouse has moved.

Notes: The default implementation simply passes this message to the next responder.

4.23.51 mouseUp(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has released the left mouse button.

Notes: The default implementation simply passes this message to the next responder.

4.23.52 moveBackward

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character backward.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed at the beginning of the former selection. `NSResponder` declares but doesn't implement this method.

4.23.53 `moveBackwardAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection backward by one element or character.

Notes: If the end being modified is the backward end, this method expands the selection; if the end being modified is the forward end, it reduces the selection. The first `moveBackwardAndModifySelection:` or `moveForwardAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the backward end becoming the mobile one if invoked first. By default, `moveLeftAndModifySelection:` is bound to the left arrow key.

`NSResponder` declares but doesn't implement this method.

4.23.54 `moveDown`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character down.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed below the beginning of the former selection. `NSResponder` declares but doesn't implement this method.

4.23.55 `moveDownAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce the top or bottom end of the selection downward by one element, character, or line (whichever is appropriate for text direction).

Notes: If the end being modified is the bottom, this method expands the selection; if the end being modified is the top, it reduces the selection. The first `moveDownAndModifySelection:` or `moveUpAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the bottom end becoming the mobile one if invoked first.

`NSResponder` declares but doesn't implement this method.

4.23.56 `moveForward`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character forward.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed at the end of the former selection. `NSResponder` declares but doesn't implement this method.

4.23.57 `moveForwardAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection forward by one element or character.

Notes: If the end being modified is the backward end, this method reduces the selection; if the end being modified is the forward end, it expands the selection. The first `moveBackwardAndModifySelection:` or `moveForwardAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the forward end becoming the mobile one if invoked first. By default, `moveRightAndModifySelection:` is bound to the right arrow key.

`NSResponder` declares but doesn't implement this method.

4.23.58 `moveLeft`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character to the left.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed at the left end of the former selection. `NSResponder` declares but doesn't implement this method.

4.23.59 `moveLeftAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection to the left (display order) by one element or character.

Notes: If the end being modified is the left end, this method expands the selection; if the end being modified is the right end, it reduces the selection. The first `moveLeftAndModifySelection:` or `moveRightAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the left end becoming the mobile one if invoked first. By default, this method is bound to the left

arrow key.

NSResponder declares but doesn't implement this method.

The essential difference between this method and the corresponding `moveBackwardAndModifySelection`: is that the latter method moves in logical order, which can differ in bidirectional text, whereas this method moves in display order.

4.23.60 `moveParagraphBackwardAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the beginning of the previous paragraph, expanding or reducing the current selection.

Notes: If the cursor is already at the beginning of a paragraph, the selection moves backward to the beginning of the previous paragraph.

NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.61 `moveParagraphForwardAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the beginning of the next paragraph, expanding or reducing the current selection.

Notes: If the cursor is already at the end of a paragraph, the selection moves forward to the end of the next paragraph.

NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.62 `moveRight`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character to the right.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed at the right end of the former selection. NSResponder declares but doesn't implement this method.

4.23.63 `moveRightAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection to the right (display order) by one element or character.

Notes: If the end being modified is the left end, this method reduces the selection; if the end being modified is the right end, it expands the selection. The first `moveLeftAndModifySelection:` or `moveRightAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the right end becoming the mobile one if invoked first. By default, this method is bound to the right arrow key.

`NSResponder` declares but doesn't implement this method.

The essential difference between this method and the corresponding `moveForwardAndModifySelection:` is that the latter method moves in logical order, which can differ in bidirectional text, whereas this method moves in display order.

4.23.64 `moveToBeginningOfDocument`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection to the first element of the document or the insertion point to the beginning.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.65 `moveToBeginningOfDocumentAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the beginning of the document, expanding or reducing the current selection.

Notes: Available in Mac OS X v10.6 and later.

4.23.66 `moveToBeginningOfLine`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection to the first element of the selected line or the insertion point to the beginning of the line.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.67 moveToBeginningOfLineAndModifySelection

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the beginning of the line, expanding or reducing the current selection.

Notes: NSResponder declares but doesn't implement this method.
Available in Mac OS X v10.6 and later.

4.23.68 moveToBeginningOfParagraph

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the insertion point to the beginning of the selected paragraph.

Notes: NSResponder declares but doesn't implement this method.

4.23.69 moveToBeginningOfParagraphAndModifySelection

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the beginning of the current paragraph, expanding or reducing the current selection.

Notes: NSResponder declares but doesn't implement this method.
Available in Mac OS X v10.6 and later.

4.23.70 moveToEndOfDocument

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection to the last element of the document or the insertion point to the end.

Notes: NSResponder declares but doesn't implement this method.

4.23.71 moveToEndOfDocumentAndModifySelection

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the end of the document, expanding or reducing the current selection.

Notes: Available in Mac OS X v10.6 and later.

4.23.72 `moveToEndOfLine`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection to the last element of the selected line or the insertion point to the end of the line.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.73 `moveToEndOfLineAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection to the last element of the selected line or the insertion point to the end of the line.

Notes: `NSResponder` declares but doesn't implement this method.

Available in Mac OS X v10.0 and later.

4.23.74 `moveToEndOfParagraph`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the insertion point to the end of the selected paragraph.

Notes: `NSResponder` declares but doesn't implement this method.

4.23.75 `moveToEndOfParagraphAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the end of the line, expanding or reducing the current selection.

Notes: `NSResponder` declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.76 `moveToLeftEndOfLine`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the left end of the line.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed at left end of the line. `NSResponder` declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.77 `moveToLeftEndOfLineAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the left end of the line, expanding or contracting the selection as required.

Notes: Available in Mac OS X v10.6 and later.

4.23.78 `moveToRightEndOfLine`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the right end of the line.

Notes: Available in Mac OS X v10.6 and later.

4.23.79 `moveToRightEndOfLineAndModifySelection`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point to the right end of the line, expanding or contracting the selection as required.

Notes: Available in Mac OS X v10.6 and later.

4.23.80 `moveUp`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one element or character up.

Notes: In text, if there is a selection it should be deselected, and the insertion point should be placed above the beginning of the former selection. `NSResponder` declares but doesn't implement this method.

4.23.81 `moveUpAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce the top or bottom end of the selection upward by one element, character, or line (whichever is appropriate for text direction).

Notes: If the end being modified is the bottom, this method reduces the selection; if the end being modified is the top, it expands the selection. The first `moveDownAndModifySelection`: or `moveUpAndModifySelection`: method in a series determines the end being modified by always expanding. Hence, this method results in the top end becoming the mobile one if invoked first.

NSResponder declares but doesn't implement this method.

4.23.82 `moveWordBackward`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one word backward.

Notes: If there is a selection it should be deselected, and the insertion point should be placed at the end of the first word preceding the former selection. NSResponder declares but doesn't implement this method.

4.23.83 `moveWordBackwardAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection backward by one whole word.

Notes: If the end being modified is the backward end, this method expands the selection; if the end being modified is the forward end, it reduces the selection. The first `moveWordBackwardAndModifySelection:` or `moveWordForwardAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the backward end becoming the mobile one if invoked first.

NSResponder declares but doesn't implement this method.

4.23.84 `moveWordForward`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one word forward, in logical order.

Notes: If there is a selection it should be deselected, and the insertion point should be placed at the beginning of the first word following the former selection. NSResponder declares but doesn't implement this method.

4.23.85 `moveWordForwardAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection forward by one whole word.

Notes: If the end being modified is the backward end, this method reduces the selection; if the end being modified is the forward end, it expands the selection. The first `moveWordBackwardAndModifySelection:`

or `moveWordForwardAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the forward end becoming the mobile one if invoked first. `NSResponder` declares but doesn't implement this method.

4.23.86 `moveWordLeft`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one word to the left, in display order.

Notes: If there is a selection it should be deselected, and the insertion point should be placed at the end of the first word to the left of the former selection. `NSResponder` declares but doesn't implement this method.

The main difference between this method and the corresponding `moveWordBackward:` method is that the latter moves in logical order, which is important in bidirectional text, whereas this method moves in display order.

4.23.87 `moveWordLeftAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection left by one whole word in display order.

Notes: If the end being modified is the left end, this method expands the selection; if the end being modified is the right end, it reduces the selection. The first `moveWordLeftAndModifySelection:` or `moveWordRightAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the left end becoming the mobile one if invoked first.

`NSResponder` declares but doesn't implement this method.

The main difference between this method and the corresponding `moveWordBackwardAndModifySelection:` method is that the latter moves in logical order, which is important in bidirectional text, whereas this method moves in display order.

4.23.88 `moveWordRight`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to move the selection or insertion point one word right.

Notes: If there is a selection it should be deselected, and the insertion point should be placed at the beginning of the first word to the right of the former selection. `NSResponder` declares but doesn't implement this

method.

The main difference between this method and the corresponding `moveWordForward:` method is that the latter moves in logical order, which is important in bidirectional text, whereas this method moves in display order.

4.23.89 `moveWordRightAndModifySelection`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to expand or reduce either end of the selection to the right by one whole word.

Notes: If the end being modified is the backward end, this method reduces the selection; if the end being modified is the forward end, it expands the selection. The first `moveWordBackwardAndModifySelection:` or `moveWordForwardAndModifySelection:` method in a series determines the end being modified by always expanding. Hence, this method results in the forward end becoming the mobile one if invoked first. `NSResponder` declares but doesn't implement this method.

The main difference between this method and the corresponding `moveWordForwardAndModifySelection:` method is that the latter moves in logical order, which is important in bidirectional text, whereas this method moves in display order.

4.23.90 `otherMouseDown(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has pressed a mouse button other than the left or right one.

Notes: The default implementation simply passes this message to the next responder.
Available in Mac OS X v10.1 and later.

4.23.91 `otherMouseDragged(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has moved the mouse with a button other than the left or right button pressed.

Notes: The default implementation simply passes this message to the next responder.
Available in Mac OS X v10.1 and later.

4.23.92 otherMouseUp(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has released a mouse button other than the left or right button.

Notes: The default implementation simply passes this message to the next responder.

4.23.93 pageDown

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver down (or back) one page in its scroll view, also moving the insertion point to the top of the newly displayed page.

Notes: NSResponder declares but doesn't implement this method.

4.23.94 pageDownAndModifySelection

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver down (or back) one page in its scroll view, also moving the insertion point to the top of the newly displayed page. The selection is expanded or contracted as required.

Notes: NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.95 pageUp

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver up (or forward) one page in its scroll view, also moving the insertion point to the top of the newly displayed page.

Notes: NSResponder declares but doesn't implement this method.

4.23.96 pageUpAndModifySelection

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver up (or forward) one page in its scroll view, also moving the insertion point to the top of the newly displayed page. The selection is expanded or contracted as necessary.

Notes: NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.97 `performMnemonic(theString as string)` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Overridden by subclasses to handle a mnemonic.

Notes: If the character code or codes in `theString` match the receiver's mnemonic, the receiver should perform the mnemonic and return true. The default implementation does nothing and returns false. Mnemonics are not supported in Mac OS X.

4.23.98 `presentError(e as NSErrorMBS)` as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Presents an error alert to the user as an application-modal dialog.

Notes: `e`: An object containing information about an error.

The alert displays information found in the `NSError` object `e`; this information can include error description, recovery suggestion, failure reason, and button titles (all localized). The method returns true if error recovery succeeded and false otherwise. For error recovery to be attempted, a recovery-attempter object (that is, an object conforming to the `NSErrorRecoveryAttempting` informal protocol) must be associated with `e`.

The default implementation of this method sends `willPresentError` to self. By doing this, `NSResponder` gives subclasses an opportunity to customize error presentation. It then forwards the message, passing any customized error object, to the next responder; if there is no next responder, it passes the error object to `NSApp`, which displays a document-modal error alert. When the user dismisses the alert, any recovery attempter associated with the error object is given a chance to recover from the error. See the class description for the precise route up the responder chain (plus document and controller objects) this message might travel.

Available in OS X v10.4 and later.

4.23.99 `rightMouseDown(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has pressed the right mouse button.

Notes: The default implementation simply passes this message to the next responder.

4.23.100 rightMouseDown(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has moved the mouse with the right button pressed.

Notes: The default implementation simply passes this message to the next responder.

4.23.101 rightMouseUp(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has released the right mouse button.

Notes: The default implementation simply passes this message to the next responder.

4.23.102 rotateWithEvent(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has begun a rotation gesture.

Notes: The event will be sent to the view under the touch in the key window.

Available in Mac OS X v10.6 and later.

4.23.103 scrollLineDown

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver one line down in its scroll view, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

4.23.104 scrollLineUp

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver one line up in its scroll view, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

4.23.105 scrollPageDown

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver one page down in its scroll view, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

4.23.106 scrollPageUp

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver one page up in its scroll view, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

4.23.107 scrollToBeginningOfDocument

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver to the beginning of the document, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.108 scrollToEndOfDocument

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to scroll the receiver to the end of the document, without changing the selection.

Notes: NSResponder declares but doesn't implement this method.

Available in Mac OS X v10.6 and later.

4.23.109 scrollWheel(e as NSEventMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the mouse's scroll wheel has moved.

Notes: The default implementation simply passes this message to the next responder.

Available in Mac OS X v10.0 and later.

4.23.110 selectAll

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to select all selectable elements.

Notes: NSResponder declares but doesn't implement this method.

4.23.111 selectLine

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to select all elements in the line or lines containing the selection or insertion point.

Notes: NSResponder declares but doesn't implement this method.

4.23.112 selectParagraph

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to select all paragraphs containing the selection or insertion point.

Notes: NSResponder declares but doesn't implement this method.

4.23.113 selectToMark

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to select all items from the insertion point or selection to a previously placed mark, including the selection itself if not empty.

Notes: NSResponder declares but doesn't implement this method.

4.23.114 selectWord

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to extend the selection to the nearest word boundaries outside it (up to, but not including, word delimiters).

Notes: NSResponder declares but doesn't implement this method.

4.23.115 setMark

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to set a mark at the insertion point or selection, which is used by deleteToMark and selectToMark.

Notes: NSResponder declares but doesn't implement this method.

4.23.116 showContextHelp

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to invoke the help system, displaying information relevant to the receiver and its current state.

4.23.117 swapWithMark

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Swaps the mark and the selection or insertion point, so that what was marked is now the selection or insertion point, and what was the insertion point or selection is now the mark.

Notes: NSResponder declares but doesn't implement this method.

4.23.118 `swipeWithEvent(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that the user has begun a rotation gesture.

Notes: The event will be sent to the view under the touch in the key window.
Available in Mac OS X v10.6 and later.

4.23.119 `tabletPoint(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that a tablet-point event has occurred.

Notes: Tablet events are represented by NSEvent objects of type NSTabletPoint. They describe the current state of a transducer (that is, a pointing device) that is in proximity to its tablet, reflecting changes such as location, pressure, tilt, and rotation. See the NSEvent reference for the methods that allow you to extract this and other information from theEvent. The default implementation of NSResponder passes the message to the next responder.

Available in Mac OS X v10.4 or later.

4.23.120 `tabletProximity(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Informs the receiver that a tablet-proximity event has occurred.

Notes: Tablet events are represented by NSEvent objects of type NSTabletProximity. Tablet devices generate proximity events when the transducer (pointing device) nears a tablet and when it moves away from a tablet. From an event object of this type you can extract information about the kind of device and its capabilities, as well as the relation of this tablet-proximity event to various tablet-point events; see the NSEvent reference for details. The default implementation passes the message to the next responder.

Available in Mac OS X v10.4 or later.

4.23.121 `transpose`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Transposes the characters to either side of the insertion point and advances the insertion point past both of them. Does nothing to a selected range of text.

Notes: NSResponder declares but doesn't implement this method.

4.23.122 transposeWords

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Transposes the words to either side of the insertion point and advances the insertion point past both of them. Does nothing to a selected range of text.

Notes: NSResponder declares but doesn't implement this method.

4.23.123 undoManager as NSUndoManagerMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The undo manager for this item.

Notes: Subclasses may implement this property.

4.23.124 uppercaseWord

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Implemented by subclasses to make uppercase every letter in the word or words surrounding the insertion point or selection, expanding the selection if necessary.

Notes: If either end of the selection partially covers a word, that entire word is made uppercase. NSResponder declares but doesn't implement this method.

4.23.125 yank

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Replaces the insertion point or selection with text from the kill buffer.

Notes: If invoked sequentially, cycles through the kill buffer in reverse order. See "Standard Action Methods for Selecting and Editing" for more information on the kill buffer. NSResponder declares but doesn't implement this method.

4.23.126 Properties

4.23.127 Handle as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop & iOS.

Function: The internal handle of this Responder.

Notes: (Read and Write property)

4.23.128 menu as NSMenuMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The receiver's menu.

Notes: (On useful for NSApplication class on Mac)
(Read and Write computed property)

4.23.129 nextResponder as NSResponderMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The next responder in the responder chain.

Notes: (Read and Write computed property)

4.24 class NSScreenMBS

4.24.1 class NSScreenMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: An NSScreen object describes the attributes of a computer's monitor, or screen.

Example:

```
Dim mainScreen As NSScreenMBS = NSScreenMBS.mainScreen
```

```
msgbox mainScreen.localizedName
```

Notes: An application may use an NSScreen object to retrieve information about a screen and use this information to decide what to display upon that screen. For example, an application may use the `deepestScreen` method to find out which of the available screens can best represent color and then may choose to display all of its windows on that screen.

The application object should be created before you use the methods in this class, so that the application object can make the necessary connection to the window system. You can make sure the application object exists by invoking the `sharedApplication` method of `NSApplication`. If you created your application with Xcode, the application object is automatically created for you during initialization.

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 19.4pr2](#)
- [Colorspaces in MacOS with Xojo](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)
- [MBS REALbasic plug-in 9.6](#)

Xojo Developer Magazine

- [7.6, page 8: News](#)

4.24.2 Methods

4.24.3 `backingAlignedRect(r as NSRectMBS, options as UInt64) as NSRectMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Converts a rectangle in global screen coordinates to a pixel aligned rectangle.

Notes: `r`: The input rectangle in global screen coordinates.

options: Specifies the alignment options. See `NSAlignmentOptions` for possible values.

Returns a a pixel aligned rectangle on the target screen from the given input rectangle in global screen coordinates.

This method uses `NSIntegralRectWithOptions()` to produce the pixel aligned rectangle.
Available in Mac OS X v10.7 and later.

4.24.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop & iOS.

Function: The private constructor.

4.24.5 `convertRectFromBacking(r as NSRectMBS) as NSRectMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Converts the rectangle from the device pixel aligned coordinates system of a screen.

Notes: r: The rectangle.

Returns the rectangle converted from the device pixel aligned coordinates system of the screen.
Available in Mac OS X v10.7 and later.

4.24.6 `convertRectToBacking(r as NSRectMBS) as NSRectMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Converts the rectangle to the device pixel aligned coordinates system of a screen.

Notes: r: The rectangle.

Returns the rectangle converted to the device pixel aligned coordinates system of the screen.
Available in Mac OS X v10.7 and later.

4.24.7 `NSScreenColorSpaceDidChangeNotification` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: The notification name for the notification sent when the screen colorspace changed.

Notes: The notification object is the `NSScreen` whose `colorSpace` has changed. This notification does not

contain a `userInfo` dictionary.

Use this constant with `NSNotificationObserverMBS` class to get an event when such a notification is sent. Available in Mac OS X v10.6 and later.

4.24.8 `screens as NSScreenMBS()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an array of `NSScreen` objects representing all of the screens available on the system.

Notes: Returns an array of the `NSScreen` objects available on the current system or `nil` if the screen information could not be obtained from the window system.

The screen at index 0 in the returned array corresponds to the primary screen of the user's system. This is the screen that contains the menu bar and whose origin is at the point (0, 0). In the case of mirroring, the first screen is the largest drawable display; if all screens are the same size, it is the screen with the highest pixel depth. This primary screen may not be the same as the one returned by the `mainScreen` method, which returns the screen with the active window.

The array should not be cached. Screens can be added, removed, or dynamically reconfigured at any time. When the display configuration is changed, the default notification center sends a `NSApplicationDidChangeScreenParametersNotification` notification.

4.24.9 `supportedWindowDepths as Integer()`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a zero-terminated array of the window depths supported by the receiver.

4.24.10 Properties

4.24.11 `backingScaleFactor as Double`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the backing store pixel scale factor for the screen.

Example:

```
Msgbox str(NSScreenMBS.mainScreen.backingScaleFactor)
```

Notes: Returns the scale factor representing the number of backing store pixels corresponding to each linear

unit in screen space on this NSScreen.

This method is provided for rare cases when the explicit scale factor is needed. Please use the NSView class's convert backing methods whenever possible.

Available in Mac OS X v10.7 and later.

For apps which are not enabled for retina support, the function returns 1. So you only see 2 here if app is Cocoa, display is retina and info.plist has the NSHighResolutionCapable key.

(Read only property)

4.24.12 colorSpace as Variant

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the colorSpace of the screen.

Example:

```
dim s as NSScreenMBS = NSScreenMBS.mainScreen
dim c as NSColorSpaceMBS = s.colorSpace
MsgBox c.localizedName
```

Notes: Available in Mac OS X v10.6 and later.

Value is a NSColorSpaceMBS but declared as Variant to reduce plugin interdependencies.

(Read only property)

4.24.13 deepestScreen as NSScreenMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns an NSScreen object representing the screen that can best represent color.

Notes: This method always returns an object, even if there is only one screen and it is not a color screen.

(Read only property)

4.24.14 depth as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's current bit depth and colorspace information.

Notes: Returns the window depth information. This value cannot be used directly. You must pass it to a function such as NSBitsPerPixelFromDepth or NSColorSpaceFromDepth to obtain a concrete value for the desired information.

(Read only property)

4.24.15 deviceDescription as dictionary

Plugin Version: 11.3, Platform: macOS, Targets: Desktop only.

Function: Returns the device dictionary for the screen.

Example:

```
Dim dic As Dictionary = NSScreenMBS.mainScreen.deviceDescription
```

```
Break // see content in debugger
```

Notes: A dictionary containing the attributes of the receiver's screen.
(Read only property)

4.24.16 firstScreen as NSScreenMBS

Plugin Version: 12.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the NSScreen object for the first screen.

Example:

```
MessageBox NSScreenMBS.firstScreen.localizedName
```

Notes: Returns nil if no display is present.
(Read only property)

4.24.17 frame as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the dimensions and location of the receiver.

Example:

```
Dim screens() As NSScreenMBS = NSScreenMBS.screens
```

```
// show frame for each screen
```

```
For Each Screen As NSScreenMBS In screens
```

```
MsgBox Screen.frame.String
```

```
Next
```

Notes: The full screen rectangle at the current resolution. This rectangle includes any space currently occupied by the menu bar and dock.
(Read only property)

4.24.18 Handle as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The internal reference to the NSScreen object.

Notes: (Read and Write property)

4.24.19 localizedName as String

Plugin Version: 19.4, Platform: macOS, Targets: Desktop & iOS.

Function: The localized name of the screen.

Example:

```
MessageBox NSScreenMBS.firstScreen.localizedName
```

Notes: Either uses on MacOS 10.15 the value Apple frameworks provides.
Or returns name from plugin, either "Main Screen" or "Screen " plus index, so second screen is "Screen 2".
(Read only property)

4.24.20 mainScreen as NSScreenMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the NSScreen object containing the window with the keyboard focus.

Example:

```
Dim mainScreen As NSScreenMBS = NSScreenMBS.mainScreen
```

```
msgbox mainScreen.localizedName
```

Notes: The main screen is not necessarily the same screen that contains the menu bar or has its origin at (0, 0). The main screen refers to the screen containing the window that is currently receiving keyboard

events. It is the main screen because it is the one with which the user is most likely interacting.

The screen containing the menu bar is always the first object (index 0) in the array returned by the `screens` method.

(Read only property)

4.24.21 `screensHaveSeparateSpaces` as Boolean

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

Function: Returns a Boolean value indicating whether each screen can have its own set of spaces.

Notes: This method reflects whether the „Displays have separate Spaces,“ option is enabled in Mission Control system preference. You might use the return value to determine how to present your app when in fullscreen mode.

(Read only property)

4.24.22 `secondScreen` as `NSScreenMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the `NSScreen` object for the second screen.

Notes: Returns nil if no second display is present.

(Read only property)

4.24.23 `userSpaceScaleFactor` as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the scaling factor from user space to device space on the screen represented by the receiver.

Example:

```
Msgbox str(NSScreenMBS.mainScreen.userSpaceScaleFactor)
```

Notes: Returns the scaling factor, measured in pixels per point, where a point is always equal to 1/72 of an inch. For example, a scaling factor of 2.0 indicates the display has a resolution 2 pixels per point or 144 pixels-per-inch.

Available in Mac OS X v10.4 and later.

(Read only property)

4.24.24 visibleFrame as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the current location and dimensions of the visible screen.

Example:

```
Dim screens() As NSScreenMBS = NSScreenMBS.screens
```

```
// shows coordinates for each screen.
```

```
For Each Screen As NSScreenMBS In screens
```

```
MsgBox Screen.visibleFrame.String
```

```
Next
```

Notes: The returned rectangle is always based on the current user-interface settings and does not include the area currently occupied by the dock and menu bar. Because it is based on the current user -interface settings, the returned rectangle can change between calls and should not be cached.

Note: Even when dock hiding is enabled, the rectangle returned by this method may be smaller than the full screen. The system uses a small boundary area to determine when it should display the dock.

(Read only property)

4.25 class NSSoundDelegateMBS

4.25.1 class NSSoundDelegateMBS

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class for the delegate to receive events from NSSoundMBS objects.

4.25.2 Events

4.25.3 SoundFinished(s as NSSoundMBS, didFinishPlaying as boolean)

Plugin Version: 9.4, Platform: macOS, Targets: .

Function: This delegate method is called when an NSSoundMBS instance has completed playback of its sound data.

Notes: s: The NSSound that has completed playback of its sound data.

didFinishPlaying: True when playback was successful; false otherwise.

The parameter s is not the NSSoundMBS object you used before but a new one. But it has the same handle value as the one where you called play, so you can still compare which one was affected.

4.26 class NSSoundMBS

4.26.1 class NSSoundMBS

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSSound class provides a simple interface for loading and playing audio files.

Notes: This class supports the same audio encodings and file formats that are supported by Core Audio and QuickTime.

To use this class, initialize a new instance with the desired file or audio data. You can configure assorted aspects of the audio playback, including the volume and whether the sound loops before you play it. Depending on the type of the audio data, this class may use either Core Audio or QuickTime to handle the actual playback. (Typically, it uses Core Audio to play files in the AIFF, WAVE, NeXT, SD2, AU, and MP3 formats and may use it for other formats in the future as well.) Playback occurs asynchronously so that your application can continue doing work.

You should retain NSSound objects before initiating playback or make sure you have a strong reference to them in a garbage-collected environment. Upon deallocation, a sound object stops playback of the sound (as needed) so that it can free up the corresponding audio resources. If you want to deallocate a sound object immediately after playback, assign a delegate and use the `sound:didFinishPlaying:` method to deallocate it.

If you want to play the system beep sound, use the `NSBeep` function.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.3pr1](#)
- [MBS REALbasic plug-ins version 9.5](#)
- [MBS REALbasic plug-ins version 9.4](#)

4.26.2 Methods

4.26.3 availableSounds as string()

Plugin Version: 9.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of the available sounds.

Example:

```
dim sounds(-1) as string = NSSoundMBS.availableSounds

// play first sound
dim n as NSSoundMBS = NSSoundMBS.soundNamed(sounds(0))

call n.play
```

```
// show list of sounds
MsgBox Join(sounds,EndOfLine)
```

Notes: You can use the names in this array for the `soundNamed` function.

4.26.4 `canInitWithPasteboard` as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver can create an instance of itself from the data in a pasteboard.

Notes: true when the receiver can handle the data represented by pasteboard; false otherwise.

The `soundUnfilteredPasteboardTypes` method is used to find out whether the class can handle the data in pasteboard.

4.26.5 `channelMapping` as `Integer()`

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the receiver,Ãs channel map.

Notes: A channel map correlates a sound,Ãs channels to the the output-device,Ãs channels. For example, a two-channel sound being played on a five-channel device should have a channel map to optimize the sound-playing experience. The default map, correlates the first sound channel to the first output channel, the second sound channel to the second output channel, and so on.

Available in OS X v10.5 and later. Deprecated in OS X v10.9.

4.26.6 `Constructor`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the receiver with data from a pasteboard.

Notes: `NSSound` expects the data to have a proper magic number, sound header, and data for the formats it supports.

See also:

- 4.26.7 `Constructor(data as MemoryBlock)` 375
- 4.26.8 `Constructor(file as folderitem, ByReference as boolean)` 375

4.26. CLASS NSSOUNDMBS	375
• 4.26.9 Constructor(url as string, ByReference as boolean)	376

4.26.7 Constructor(data as MemoryBlock)

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the receiver with a given audio data.

Notes: data: Audio data with which the receiver is to be initialized. The data must have a proper magic number, sound header, and data for the formats the NSSound class supports.

See also:

• 4.26.6 Constructor	374
• 4.26.8 Constructor(file as folderitem, ByReference as boolean)	375
• 4.26.9 Constructor(url as string, ByReference as boolean)	376

4.26.8 Constructor(file as folderitem, ByReference as boolean)

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the receiver with the the audio data located at a given file.

Example:

```
dim s as NSSoundMBS
```

```
dim f as FolderItem
```

```
f=SpecialFolder.Desktop.Child("test.mp3")
```

```
s=new NSSoundMBS(f,true)
```

```
call s.play
```

Notes: file: Path to the sound file.

ByReference: When true only the name of the sound is stored with the NSSound instance when archived using encodeWithCoder;; otherwise the audio data is archived along with the instance. (not used in Xojo)

See also:

• 4.26.6 Constructor	374
• 4.26.7 Constructor(data as MemoryBlock)	375
• 4.26.9 Constructor(url as string, ByReference as boolean)	376

4.26.9 Constructor(url as string, ByReference as boolean)

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the receiver with the audio data located at a given URL.

Notes: url: URL to the sound file with which the receiver is to be initialized.

ByReference: When true only the name of the sound is stored with the NSSound instance when archived using encodeWithCoder::; otherwise the audio data is archived along with the instance. (not used in Xojo)

See also:

- 4.26.6 Constructor 374
- 4.26.7 Constructor(data as MemoryBlock) 375
- 4.26.8 Constructor(file as folderitem, ByReference as boolean) 375

4.26.10 duration as Double

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the duration of the receiver in seconds.

Notes: Available in Mac OS X v10.5 and later.

4.26.11 isPlaying as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver is playing its audio data.

Notes: True when the receiver is playing its audio data, false otherwise.

4.26.12 name as string

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the name assigned to the receiver.

Notes: Name assigned to the receiver; "" when no name has been assigned.

4.26.13 NSSoundPboardType as string

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSSound class defines this common pasteboard data type.

4.26.14 pause as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Pauses audio playback.

Notes: True when playback is paused successfully, false when playback is already paused or when an error occurred.

4.26.15 play as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initiates audio playback.

Notes: True when playback is initiated, false when playback is already in progress or when an error occurred.

This method initiates playback asynchronously and returns control to your application. Therefore, your application can continue doing work while the audio is playing.

4.26.16 resume as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Resumes audio playback.

Notes: True when playback is resumed, false when playback is in progress or when an error occurred.

Assumes the receiver has been previously paused by sending it pause.

4.26.17 setChannelMapping(mapping() as Integer)

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Specifies the receiver's channel map.

Notes: Mapping: Audio-channel—to—device-channel mappings for the receiver.

Available in OS X v10.5 and later. Deprecated in OS X v10.9.

4.26.18 setDelegate(DelegateHandler as NSSoundDelegateMBS)

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Set the receiver's delegate.

Notes: You can assign one delegate to several sounds.

But the delegate object is not referenced, so keep it alive with your own reference, so RB won't destroy it too early.

4.26.19 setName(name as string) as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Registers the receiver under a given name.

Notes: Returns True when successful; false otherwise.

If the receiver is already registered under another name, this method first unregisters the prior name.

4.26.20 soundNamed(name as string) as NSSoundMBS

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the NSSound instance associated with a given name.

Example:

```
dim s as NSSoundMBS
```

```
s=NSSoundMBS.soundNamed("Submarine")
```

```
call s.play
```

Notes: NSSound instance initialized with the sound data identified by soundName.

The returned object can be one of the following:

One that's been assigned a name with setName.

One of the named system sounds provided by the Application Kit framework. If there's no known NSSound object with soundName, this method tries to create one by searching for sound files in the application's main bundle (see NSBundle for a description of how the bundle's contents are searched). If no sound file can be located in the application main bundle, the following directories are searched in order:

textasciitilde /Library/Sounds, /Library/Sounds, /Network/Library/Sounds or /System/Library/Sounds.

If no data can be found for soundName, no object is created, and nil is returned.

The preferred way to locate a sound is to pass a name without the file extension. See the class description for a list of the supported sound file extensions.

4.26.21 soundUnfilteredFileTypes as string()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the list of file types the NSSound class understands.

Notes: Returns array of strings representing the file types the NSSound class understands.

The returned array may be passed directly to the runModalForTypes method of the NSOpenPanel class.

Available in OS X v10.0 and later. Deprecated in OS X v10.5.

4.26.22 soundUnfilteredPasteboardTypes as string()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides a list of the pasteboard types that the NSSound class can accept.

Notes: Array of pasteboard types that the NSSound class can accept.

Available in OS X v10.0 and later. Deprecated in OS X v10.5.

4.26.23 soundUnfilteredTypes as string()

Plugin Version: 14.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the file types the NSSound class understands.

Notes: Returns array of UTIs identifying the file types the NSSound class understands.

Available in OS X v10.5 and later.

4.26.24 soundWithContentsOfFile(file as folderitem, ByReference as boolean) as NSSoundMBS

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a NSSound object with the the audio data located at a given file.

Example:

```
dim s as NSSoundMBS
```

```
dim f as FolderItem
```

```
f=SpecialFolder.Desktop.Child("test.mp3")
s=NSSoundMBS.soundWithContentsOfFile(f,true)
call s.play

// sound continues to play even after NSSoundMBS object is destroyed.
```

Notes: file: Path to the sound file.

ByReference: When true only the name of the sound is stored with the NSSound instance when archived using `encodeWithCoder::`; otherwise the audio data is archived along with the instance. (not used in Xojo)

4.26.25 `soundWithContentsOfURL(url as string, ByReference as boolean)` as `NSSoundMBS`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a NSSoundMBS object with the audio data located at a given URL.

Notes: url: URL to the sound file with which the receiver is to be initialized.

ByReference: When true only the name of the sound is stored with the NSSound instance when archived using `encodeWithCoder::`; otherwise the audio data is archived along with the instance. (not used in Xojo)

4.26.26 `soundWithData(data as MemoryBlock)` as `NSSoundMBS`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a NSSoundMBS object with a given audio data.

Notes: data: Audio data with which the receiver is to be initialized. The data must have a proper magic number, sound header, and data for the formats the NSSound class supports.

4.26.27 `soundWithPasteboard` as `NSSoundMBS`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a NSSoundMBS object with data from a pasteboard.

Notes: NSSound expects the data to have a proper magic number, sound header, and data for the formats it supports.

4.26.28 stop as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Concludes audio playback.

Notes: True when playback is concluded successfully or if it's paused, false otherwise.

4.26.29 writeToPasteboard

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes the receiver's data to a pasteboard.

4.26.30 Properties

4.26.31 Handle as Integer

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal reference to the NSSound object.

Notes: (Read and Write property)

4.26.32 currentTime as Double

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the receiver's playback progress in seconds.

Notes: Receiver's playback progress in seconds.

Sounds start with `currentTime == 0` and end with `currentTime == ([<sound>duration] - 1)`.

Available in Mac OS X v10.5 and later.

This property is not archived, copied, or stored on the pasteboard.

(Read and Write computed property)

4.26.33 loops as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Indicates whether the receiver restarts playback when it reaches the end of its content.

Notes: True when the receiver restarts playback when it finishes, false otherwise.

Default: false

Available in Mac OS X v10.5 and later.

When loops is true, the delegate does not call the SoundFinished event on the end of its content and restarts playback.

(Read and Write computed property)

4.26.34 playbackDeviceIdentifier as string

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Identifies the receiver's output device.

Notes: Returns an unique identifier of a sound output device.

Available in Mac OS X v10.5 and later.

(Read and Write computed property)

4.26.35 volume as Double

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Provides the volume of the receiver.

Notes: This method does not affect the systemwide volume.

Available in Mac OS X v10.5 and later.

0.0 is not sound and 1.0 is full sound.

(Read and Write computed property)

4.27 class NSStreamMBS

4.27.1 class NSStreamMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: An abstract class representing a stream.

Notes: This class's interface is common to all Cocoa stream classes, including its concrete subclasses `NSInputStream` and `NSOutputStream`.

`NSStream` objects provide an easy way to read and write data to and from a variety of media in a device-independent way. You can create stream objects for data located in memory, in a file, or on a network (using sockets), and you can use stream objects without loading all of the data into memory at once.

By default, `NSStream` instances that are not file-based are non-seekable, one-way streams (although custom seekable subclasses are possible). Once the data has been provided or consumed, the data cannot be retrieved from the stream.

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 18.2pr2](#)

4.27.2 Methods

4.27.3 Close

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Closes the receiver.

Notes: Closing the stream terminates the flow of bytes and releases system resources that were reserved for the stream when it was opened. If the stream has been scheduled on a run loop, closing the stream implicitly removes the stream from the run loop. A stream that is closed can still be queried for its properties.

4.27.4 Constructor

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The private constructor.

4.27.5 Open

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Opens the receiving stream.

Notes: A stream must be created before it can be opened. Once opened, a stream cannot be closed and

reopened.

4.27.6 SetPosition(pos as Int64) as boolean

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Sets position in a stream.

Notes: Returns true on success or false on failure.

4.27.7 Properties

4.27.8 Error as NSErrorMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Returns an NSError object representing the stream error.

Notes: An NSError object representing the stream error, or nil if no error has been encountered.
(Read only property)

4.27.9 Handle as Integer

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.27.10 position as Int64

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Queries position in file stream.

Notes: (Read only property)

4.27.11 Status as Integer

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Returns the receiver's status.

Notes: See kStatus* constants.

(Read only property)

4.27.12 Constants

Status Constants

Constant	Value	Description
kStatusAtEnd	5	There is no more data to read, or no more data can be written to the stream. When this status is returned, the stream is in a ,Änon-blocking,Ä mode and no data are available.
kStatusClosed	6	The stream is closed (close has been called on it).
kStatusError	7	The remote end of the connection can,Ät be contacted, or the connection has been severed for some other reason.
kStatusNotOpen	0	The stream is not open for reading or writing. This status is returned before the underlying call to open a stream but after it,Äs been created.
kStatusOpen	2	The stream is open, but no reading or writing is occurring.
kStatusOpening	1	The stream is in the process of being opened for reading or for writing. For network streams, this status might include the time after the stream was opened, but while network DNS resolution is happening.
kStatusReading	3	Data is being read from the stream. This status would be returned if code on another thread were to call Status on the stream while a read() was in progress.
kStatusWriting	4	Data is being written to the stream. This status would be returned if code on another thread were to call Status on the stream while a write() was in progress.

4.28 class NSTimeZoneMBS

4.28.1 class NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: NSTimeZone is an abstract class that defines the behavior of time zone objects.

Notes: Time zone objects represent geopolitical regions. Consequently, these objects have names for these regions. Time zone objects also represent a temporal offset, either plus or minus, from Greenwich Mean Time (GMT) and an abbreviation (such as PST for Pacific Standard Time).

NSTimeZone provides several class methods to get time zone objects: `timeZoneWithName`, `timeZoneWithAbbreviation`, and `timeZoneForSecondsFromGMT`. The class also permits you to set the default time zone within your application (`setDefaultTimeZone`). You can access this default time zone at any time with the `defaultTimeZone` class method, and with the `localTimeZone` class method, you can get a relative time zone object that decodes itself to become the default time zone for any locale in which it finds itself.

Cocoa does not provide any API to change the time zone of the computer, or of other applications.

Some `NSDate` methods return date objects that are automatically bound to time zone objects. These date objects use the functionality of `NSTimeZone` to adjust dates for the proper locale. Unless you specify otherwise, objects returned from `NSDate` are bound to the default time zone for the current locale.

Note that, strictly, time zone database entries such as "America/Los_Angeles" are IDs not names. An example of a time zone name is "Pacific Daylight Time". Although many `NSTimeZone` method names include the word "name", they refer to IDs.

Blog Entries

- [MBS Xojo Plugins, version 17.5pr4](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr4](#)
- [MBS Real Studio Plugins, version 12.3pr11](#)

4.28.2 Methods

4.28.3 Constructor

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Creates a new timezone.

See also:

- 4.28.4 `Constructor(name as string)`

4.28.4 Constructor(name as string)

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Creates a time zone initialized with a given ID.

Example:

```
dim t as NSTimeZoneMBS = new NSTimeZoneMBS("Europe/Berlin")
MsgBox t.name
```

Notes: If name is a known ID, this method calls initWithName:data: with the appropriate data object.

In Mac OS X v10.4 and earlier providing nil for the parameter would have caused a crash. In Mac OS X v10.5 and later, this now raises an invalid argument exception.

See also:

- 4.28.3 Constructor

386

4.28.5 copy as NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Creates a copy of the timezone object.

Example:

```
dim t as NSTimeZoneMBS = NSTimeZoneMBS.systemTimeZone
dim c as NSTimeZoneMBS = t.copy
```

```
MsgBox c.name
```

4.28.6 defaultTimeZone as NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the default time zone for the current application.

Example:

```
MsgBox NSTimeZoneMBS.defaultTimeZone.name
```

Notes: The default time zone for the current application. If no default time zone has been set, this method invokes systemTimeZone and returns the system time zone.

Discussion

The default time zone is the one that the application is running with, which you can change (so you can make the application run as if it were in a different time zone).

If you get the default time zone and hold onto the returned object, it does not change if a subsequent invocation of `setDefaultTimeZone` changes the default time zone—you still have the specific time zone you originally got. Contrast this behavior with the object returned by `localTimeZone`.

4.28.7 `isEqualToTimeZone(timeZone as NSTimeZoneMBS)` as boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver has the same name and data as another given time zone.

Example:

```
if NSTimeZoneMBS.systemTimeZone.isEqualToTimeZone(NSTimeZoneMBS.localTimeZone) then
  MsgBox "equal"
else
  MsgBox "not equal"
end if
```

Notes: `TimeZone`: The time zone to compare with the receiver.

Returns true if `TimeZone` and the receiver have the same name and data, otherwise false.

4.28.8 `knownTimeZoneNames` as string()

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns an array of strings listing the IDs of all the time zones known to the system.

Example:

```
MsgBox join(NSTimeZoneMBS.knownTimeZoneNames, EndOfLine)
```

4.28.9 `localTimeZone` as NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns an object that forwards all messages to the default time zone for the current application.

Example:

MsgBox NSTimeZoneMBS.localTimeZone.name

Notes: An object that forwards all messages to the default time zone for the current application.

Discussion

The local time zone represents the current state of the default time zone at all times. If you get the default time zone (using `defaultTimeZone`) and hold onto the returned object, it does not change if a subsequent invocation of `setDefaultTimeZone` changes the default time zone—you still have the specific time zone you originally got. The local time zone adds a level of indirection, it acts as if it were the current default time zone whenever you invoke a method on it.

4.28.10 Print

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Writes the time zone to debug output.

Example:

```
NSTimeZoneMBS.localTimeZone.print
```

Notes: This may help for debugging and you see output in console app.

4.28.11 systemTimeZone as NSTimeZoneMBS

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the time zone currently used by the system.

Example:

```
MsgBox NSTimeZoneMBS.systemTimeZone.name
```

Notes: The time zone currently used by the system. If the current time zone cannot be determined, returns the GMT time zone.

Special Considerations

If you get the system time zone, it is cached by the application and does not change if the user subsequently changes the system time zone. The next time you invoke `systemTimeZone`, you get back the same time zone you originally got. You have to invoke `resetSystemTimeZone` to clear the cached object.

4.28.12 timeZoneForSecondsFromGMT(seconds as Integer) as NSTimeZoneMBS

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

Function: Creates time zone object based on a delta from GMT.

Notes: Time zones created with this never have daylight savings and the offset is constant no matter the date; the name and abbreviation do NOT follow the POSIX convention (of minutes-west).

4.28.13 timeZoneWithName(name as string) as NSTimeZoneMBS

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns the time zone object identified by a given ID.

Example:

```
dim t as NSTimeZoneMBS = NSTimeZoneMBS.timeZoneWithName("Europe/Berlin")
MsgBox t.name
```

Notes: Name: The ID for the time zone.

Returns the time zone in the information directory with a name matching name.

Returns nil if there is no match for the name.

4.28.14 Properties**4.28.15 abbreviation as string**

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the abbreviation for the receiver.

Example:

```
MsgBox NSTimeZoneMBS.localTimeZone.Abbreviation
```

Notes: The abbreviation for the receiver, such as "EDT" (Eastern Daylight Time).

Invokes `abbreviationForDate` with the current date as the argument.

(Read only property)

4.28.16 DaylightSavingTimeOffset as Double

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns the current daylight saving time offset of the receiver.

Example:

```
MsgBox str(NSTimeZoneMBS.localTimeZone.DaylightSavingTimeOffset)
```

Notes: Available in OS X v10.5 and later.

(Read only property)

4.28.17 description as string

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the description of the receiver.

Example:

```
MsgBox NSTimeZoneMBS.localTimeZone.Description
```

Notes: The description of the receiver, including the name, abbreviation, offset from GMT, and whether or not daylight savings time is currently in effect.

(Read only property)

4.28.18 Handle as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.28.19 isDaylightSavingTime as Boolean

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns a Boolean value that indicates whether the receiver is currently using daylight saving time.

Example:

```
MsgBox str(NSTimeZoneMBS.localTimeZone.isDaylightSavingTime)
```

Notes: Returns true if the receiver is currently using daylight savings time, otherwise false.
(Read only property)

4.28.20 name as string

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Returns the geopolitical region ID that identifies the receiver.

Example:

```
MsgBox NSTimeZoneMBS.localTimeZone.Name
```

Notes: (Read only property)

4.28.21 SecondsFromGMT as Double

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns the current difference in seconds between the receiver and Greenwich Mean Time.

Example:

```
MsgBox str(NSTimeZoneMBS.localTimeZone.SecondsFromGMT)
```

Notes: The current difference in seconds between the receiver and Greenwich Mean Time.
(Read only property)

4.28.22 abbreviationDictionary as Dictionary

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns a dictionary holding the mappings of time zone abbreviations to time zone names.

Notes: Note that more than one time zone may have the same abbreviation—for example, US/Pacific and Canada/Pacific both use the abbreviation "PST." In these cases, abbreviationDictionary chooses a single name to map the abbreviation to.

(Read and Write computed property)

4.29 class NSTouchMBS

4.29.1 class NSTouchMBS

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: An object representing the location, size, movement, and force of a touch occurring on the screen.

Notes: You access touch objects through UIEvent objects passed into responder objects for event handling. A touch object includes accessors for:

- The view or window in which the touch occurred
- The location of the touch within the view or window
- The approximate radius of the touch
- The force of the touch (on devices that support 3D Touch or Apple Pencil)

A touch object also contains a timestamp indicating when the touch occurred, an integer representing the number of times the user tapped the screen, and the phase of the touch in the form of a constant that describes whether the touch began, moved, or ended, or whether the system canceled the touch.

To learn how to work with swipes, read [Handling Swipe and Drag Gestures in Event Handling Guide for UIKit Apps](#).

A touch object persists throughout a multi-touch sequence. You may store a reference to a touch while handling a multi-touch sequence, as long as you release that reference when the sequence ends. If you need to store information about a touch outside of a multi-touch sequence, copy that information from the touch.

Using NSTouch on macOS and UITouch on iOS.

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 24.1pr4](#)
- [News from the MBS Xojo Plugins Version 23.5](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5](#)
- [MBS Xojo Plugins, version 23.5pr7](#)

Xojo Developer Magazine

- [22.1, page 9: News](#)

4.29.2 Methods

4.29.3 Constructor

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The private constructor.

4.29.4 `locationInView(View as NSViewMBS) as NSPointMBS`

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns the current location of the touch in the coordinate system of the given view.

Notes: `view`: The view object in whose coordinate system you want the touch located. A custom view that is handling the touch may specify `self` to get the touch location in its own coordinate system. Pass `nil` to get the touch location in the window's coordinates.

Returns a point specifying the location of the receiver in view.

This method returns the current location of a `UITouch` object in the coordinate system of the specified view. Because the touch object might have been forwarded to a view from another view, this method performs any necessary conversion of the touch location to the coordinate system of the specified view.

4.29.5 `previousLocationInView(View as NSViewMBS) as NSPointMBS`

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: Returns the previous location of the touch in the coordinate system of the given view.

Notes: `view`: The view object in whose coordinate system you want the touch located. A custom view that is handling the touch may specify `self` to get the touch location in its own coordinate system. Pass `nil` to get the touch location in the window's coordinates.

This method returns the previous location of a `UITouch` object in the coordinate system of the specified view. Because the touch object might have been forwarded to a view from another view, this method performs any necessary conversion of the touch location to the coordinate system of the specified view.

4.29.6 Properties

4.29.7 deviceSize as NSSizeMBS

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: The range of the touch device in points, such as 72 ppi.

Notes: The lower-left corner of the surface is considered (0,0).

(Read only property)

4.29.8 force as Double

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The force of the touch, where a value of 1.0 represents the force of an average touch (predetermined by the system, not user-specific).

Notes: This property is available on devices that support 3D Touch or Apple Pencil. To check at runtime if a device supports 3D Touch, read the value of the forceTouchCapability property on the trait collection for any object in your app with a trait environment.

The force reported by Apple Pencil is measured along the axis of the pencil. If you want a force perpendicular to the device, you need to calculate this value using the altitudeAngle value.

The force reported by Apple Pencil is estimated at first, and may not always be updated. To determine if an update is expected, consult estimatedPropertiesExpectingUpdates and look for a UITouchPropertyForce flag. In this scenario, estimationUpdateIndex contains a non-nil value, which you can correlate with the original touch when the update occurs. When there are no expected force updates, the entire touch sequence usually won't have updates, so it may be appropriate to apply a custom, tool-specific force curve to the touch sequence.

(Read only property)

4.29.9 Handle as Integer

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The internal object reference.

Notes: (Read and Write property)

4.29.10 identity as MemoryBlock

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: The changes to a particular touch during its lifetime.

Notes: While touch identities may be re-used, they are unique during the life of the touch, even when

multiple devices are present.

As of current macOS releases, this value is a 16 byte value, which MBS Plugin provides back as Memory-Block.

(Read only property)

4.29.11 majorRadius as Double

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The radius (in points) of the touch.

Notes: Use the value in this property to determine the size of the touch that was reported by the hardware. This value is an approximation of the size and can vary by the amount specified in the majorRadiusTolerance property.

(Read only property)

4.29.12 majorRadiusTolerance as Double

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The tolerance (in points) of the touch,Ãs radius.

Notes: This value determines the accuracy of the value in the majorRadius property. Add this value to the radius to get the maximum touch radius. Subtract the value to get the minimum touch radius.

(Read only property)

4.29.13 maximumPossibleForce as Double

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The maximum possible force for a touch.

Notes: The value of this property is sufficiently high to provide a wide dynamic range for values of the force property.

This property is available on devices that support 3D Touch or Apple Pencil. To check at runtime if a device supports 3D Touch, read the value of the forceTouchCapability property on the trait collection for any object in your app with a trait environment.

(Read only property)

4.29.14 normalizedPosition as NSPointMBS

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: The normalized position of the touch.

Notes: The normalized position is a scaled value between (0.0) and (1.0,1.0), where (0.0,0.0) is the lower-left position on the touch device.

(Read only property)

4.29.15 phase as Integer

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The phase of the touch.

Notes: The property value is a constant that indicates whether the touch began, moved, ended, or was canceled. For descriptions of the possible values of this property, see UITouchPhase.

(Read only property)

4.29.16 resting as Boolean

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: The indicator for a resting touch.

Notes: Resting touches occur when a user simply rests their thumb on the trackpad device.

(Read only property)

4.29.17 tapCount as Integer

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The number of times the finger was tapped for this given touch.

Notes: The value of this property is an integer containing the number of taps that occurred for this touch within a predefined period of time. Use this property to evaluate whether the user single-tapped, double-tapped, or even triple-tapped a particular view or window.

(Read only property)

4.29.18 timestamp as Double

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The time when the touch occurred or when it was last mutated.

Notes: The value of this property is the time, in seconds since system startup, that the touch originated or was last changed. You can store the value of this property and compare it to the timestamp in subsequent UITouch objects to determine the duration of the touch and, if it is being swiped, the speed of movement. For a definition of the time since system startup, see the description of the systemUptime method of the

NSProcessInfo class.
(Read only property)

4.29.19 type as Integer

Plugin Version: 23.5, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: The type of the touch.

Notes: Type constants on macOS and iOS are different. Please check Apple's documentation.
(Read only property)

4.29.20 view as NSViewMBS

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The view to which touches are being delivered, if any.

Notes: The value of this property is the view object to which touches are being delivered, which is not necessarily the view the touch is currently in. For example, when a gesture recognizer recognizes the touch, this property is nil because no view is receiving the touch.
(Read only property)

4.29.21 window as NSWindowMBS

Plugin Version: 23.5, Platform: iOS, Targets: iOS only.

Function: The window in which the touch initially occurred.

Notes: The value of the property is the window in which the touch originally occurred. This window might not be the same window that currently contains the touch.
(Read only property)

4.29.22 Constants

Touch Types

Constant	Value	Description
TouchTypeDirect	0	A touch resulting from direct contact with the screen.
TouchTypeIndirect	1	A touch that doesn't result from contact with the screen. Indirect touches are generated by touch input devices that are separate from the screen. For example, the trackpad of an Apple TV remote generates indirect touches.
TouchTypeIndirectPointer	3	A touch resulting from a button-based, indirect input device that describes the input sequence from button press to button release.
TouchTypePencil	2	A touch from Apple Pencil.
TouchTypeStylus	2	A pencil touch occurs when Apple Pencil interacts with the device's screen. A touch from a stylus.

4.30 class NSUserDefaultsMBS

4.30.1 class NSUserDefaultsMBS

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The Cocoa class to handle preferences.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.setStringValue "test", "Hello World"
MsgBox u.stringForKey("test")
```

Notes: The NSUserDefaults class provides a programmatic interface for interacting with the defaults system. The defaults system allows an application to customize its behavior to match a user's preferences. For example, you can allow users to determine what units of measurement your application displays or how often documents are automatically saved. Applications record such preferences by assigning values to a set of parameters in a user's defaults database. The parameters are referred to as defaults since they're commonly used to determine an application's default state at startup or the way it acts by default.

At runtime, you use an NSUserDefaults object to read the defaults that your application uses from a user's defaults database. NSUserDefaults caches the information to avoid having to open the user's defaults database each time you need a default value. The synchronize method, which is automatically invoked at periodic intervals, keeps the in-memory cache in sync with a user's defaults database.

The NSUserDefaults class provides convenience methods for accessing common types such as floats, doubles, integers, Booleans, and URLs. A default object must be a property list, that is, an instance of (or for collections a combination of instances of): NSData, NSString, NSNumber, NSDate, NSArray, or NSDictionary. If you want to store any other type of object, you should typically archive it to create an instance of NSData.

Values returned from NSUserDefaults are immutable, even if you set a mutable object as the value. For example, if you set a mutable string as the value for "MyStringDefault", the string you later retrieve using stringForKey will be immutable.

A defaults database is created automatically for each user. The NSUserDefaults class does not currently support per-host preferences. To do this, you must use the CFPreferencesMBS class. However, NSUserDefaults correctly reads per-host preferences, so you can safely mix CFPreferencesMBS code with NSUserDefaultsMBS code.

If your application supports managed environments, you can use an NSUserDefaults object to determine which preferences are managed by an administrator for the benefit of the user. Managed environments correspond to computer labs or classrooms where an administrator or teacher may want to configure the systems in a particular way. In these situations, the teacher can establish a set of default preferences and

force those preferences on users. If a preference is managed in this manner, applications should prevent users from editing that preference by disabling any appropriate controls.

The NSUserDefaults class is thread-safe.

Blog Entries

- [Important note for Webkit enabled applications in the Mac App Store](#)
- [MBS Plugins 10.3 Release Notes](#)
- [MBS REALbasic Plugins, version 10.3pr6](#)

4.30.2 Methods

4.30.3 addSuiteNamed(suiteName as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Inserts the specified domain name into the receiver's search list.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.addSuiteNamed "testsuite"
```

Notes: suiteName: The domain name to insert. This domain is inserted after the application domain.

The suiteName domain is similar to a bundle identifier string, but is not tied to a particular application or bundle. A suite can be used to hold preferences that are shared between multiple applications.

4.30.4 arrayForKey(key as string) as Variant()

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns the array associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim t(-1) as Variant
t.Append "Hello"
t.Append "World"

u.setArrayValue "test", t

dim a(-1) as Variant = u.ArrayForKey("test")
```

```
for each v as Variant in a
MsgBox v
next
```

Notes: The array associated with the specified key, or an empty array if the key does not exist or its value is not an array.

4.30.5 Constructor

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Creates a new user defaults object initialized with the defaults for the current user account.

Example:

```
dim u as new NSUserDefaultsMBS
u.setURLValue "test", "http://www.apple.de/"
MsgBox u.URLForKey("test")
```

Notes: This method does not put anything in the search list.

See also:

- 4.30.6 Constructor(username as string)

402

4.30.6 Constructor(username as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Creates an NSUserDefaultsMBS object initialized with the defaults for the specified user account.

Notes: username: The name of the user account.

Returns an initialized NSUserDefaultsMBS object whose argument and registration domains are already set up. If the current user does not have access to the specified user account, the object is invalid.

This method does not put anything in the search list. Invoke it only if you've allocated your own NSUserDefaults instance instead of using the shared one.

You do not normally use this method to initialize an instance of NSUserDefaults. Applications used by a superuser might use this method to update the defaults databases for a number of users. The user who started the application must have appropriate access (read, write, or both) to the defaults database of the

new user, or this method returns nil.

See also:

- 4.30.5 Constructor

4.30.7 dictionaryRepresentation as dictionary

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a dictionary that contains a union of all key-value pairs in the domains in the search list.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim d as Dictionary = u.dictionaryRepresentation
```

```
break // "Look here in the dictionary in the debugger."
```

Notes: Returns a dictionary containing the keys. The keys are names of defaults and the value corresponding to each key is a property list object as Variant.

As with variantForKey, key-value pairs in domains that are earlier in the search list take precedence. The combined result does not preserve information about which domain each entry came from.

4.30.8 NSArgumentDomain as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: One of the constants to specify the domain.

Notes: The domain consisting of defaults parsed from the application's arguments. These are one or more pairs of the form -default value included in the command-line invocation of the application.

4.30.9 NSGlobalDomain as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: One of the constants to specify the domain.

Notes: The domain consisting of defaults meant to be seen by all applications.

4.30.10 NSRegistrationDomain as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: One of the constants to specify the domain.

Notes: The domain consisting of a set of temporary defaults whose values can be set by the application to ensure that searches will always be successful.

4.30.11 NSUserDefaultsDidChangeNotification as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The notification name for a did change notification.

Notes: Use with the NSNotification* objects.

4.30.12 objectIsForcedForKey(key as string) as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the specified key is managed by an administrator.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
MsgBox str(u.objectIsForcedForKey("test"))
```

Notes: key: The key whose status you want to check.

Returns true if the value of the specified key is managed by an administrator, otherwise false.

This method assumes that the key is a preference associated with the current user and application. For managed keys, the application should disable any user interface that allows the user to modify the value of key.

Available in Mac OS X v10.2 and later.

See also:

- 4.30.13 objectIsForcedForKey(key as string, domain as string) as boolean

404

4.30.13 objectIsForcedForKey(key as string, domain as string) as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the key in the specified domain is managed by an administrator.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
MsgBox str(u.objectIsForcedForKey("test", u.NSGlobalDomain))
```

Notes: key: The key whose status you want to check.

domain: The domain of the key.

Returns true if the key is managed by an administrator in the specified domain, otherwise false.

This method assumes that the key is a preference associated with the current user. For managed keys, the application should disable any user interface that allows the user to modify the value of key.

Available in Mac OS X v10.2 and later.

See also:

- 4.30.12 objectIsForcedForKey(key as string) as boolean

404

4.30.14 persistentDomainForName(domainName as string) as dictionary

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns a dictionary containing the keys and values in the specified persistent domain.

Notes: Returns a dictionary containing the keys. The keys are names of defaults and the value corresponding to each key is a property list object (NSData, NSString, NSNumber, NSDate, NSArray, or NSDictionary).

4.30.15 persistentDomainNames as string()

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns an array of the current persistent domain names.

Notes: You can get the keys and values for each domain by passing the returned domain names to the persistentDomainForName method.

4.30.16 registerDefaults(dic as dictionary)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Adds the contents the specified dictionary to the registration domain.

Notes: dic: The dictionary of keys and values you want to register.

If there is no registration domain, one is created using the specified dictionary, and `NSRegistrationDomain` is added to the end of the search list.

The contents of the registration domain are not written to disk; you need to call this method each time your application starts. You can place a plist file in the application's Resources directory and call `registerDefaults` with the contents that you read in from that file.

4.30.17 `removeObjectForKey(defaultName as string)`

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Removes the value of the specified default key in the standard application domain.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.SetBoolValue("test", true)
u.removeObjectForKey "test"
MsgBox str(u.boolForKey("test")) // shows false as key does not exist
```

Notes: defaultName: The key whose value you want to remove.

Removing a default has no effect on the value returned by the `variantForKey` method if the same key exists in a domain that precedes the standard application domain in the search list.

4.30.18 `removePersistentDomainForName(domainName as string)`

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Removes the contents of the specified persistent domain from the user's defaults.

Notes: domainName: The domain whose keys and values you want. This value should be equal to your application's bundle identifier.

When a persistent domain is changed, an `NSUserDefaultsDidChangeNotification` is posted.

4.30.19 removeSuiteNamed(suiteName as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Removes the specified domain name from the receiver's search list.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.removeSuiteNamed "testsuite"
```

Notes: suiteName: The domain name to remove.

4.30.20 removeVolatileDomainForName(domainName as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Removes the specified volatile domain from the user's defaults.

Notes: domainName: The volatile domain you want to remove.

4.30.21 resetStandardUserDefaults

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Synchronizes any changes made to the shared user defaults object and releases it from memory.

Example:

```
NSUserDefaultsMBS.resetStandardUserDefaults
```

Notes: A subsequent invocation of standardUserDefaults creates a new shared user defaults object with the standard search list.

4.30.22 setArrayValue(key as string, values() as Variant)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified array.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim t(-1) as Variant
t.Append "Hello"
```

```

t.Append "World"

u.setArrayValue "test", t

dim a(-1) as Variant = u.ArrayForKey("test")

for each v as Variant in a
  MsgBox v
next

```

4.30.23 setBoolValue(key as string, value as boolean)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified Boolean value.

Example:

```

dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.setBoolValue("test", true)
MsgBox str(u.boolForKey("test"))

```

4.30.24 setDataValue(key as string, value as memoryblock)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified binary string.

Example:

```

dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults

u.setDataValue("test", "Hello √§√√°")

MsgBox u.dataForKey("test") // shows no umlauts as encoding is away in binary string

```

Notes: setStringValue is for text with encoding and setDataValue is for strings containing binary data.

4.30.25 setDictionaryValue(key as string, value as dictionary)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified dictionary.

Example:

```
dim u as new NSUserDefaultsMBS
dim d as new Dictionary

d.Value("First")="Hello"
d.Value("Second")=1234.5678
u.setDictionaryValue "test", d

dim e as Dictionary = u.dictionaryForKey("test")
for each key as Variant in e.keys
dim value as Variant = e.Value(key)
MsgBox key+": "+value
next
```

4.30.26 setDoubleValue(key as string, value as Double)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the double value.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.setDoubleValue("test", 5)
MsgBox str(u.DoubleForKey("test"))
```

Notes: Available in Mac OS X v10.5 and later. On Mac OS X 10.4 the plugin will call setFloatValue for you to avoid an exception.

4.30.27 setFileValue(key as string, value as folderitem)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified URL.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim f as FolderItem = SpecialFolder.Desktop

// write
u.setFileValue "test", f
```

```
// read  
MsgBox u.fileForKey("test").NativePath
```

Notes: Available in Mac OS X v10.6 and later.

4.30.28 setFloatValue(key as string, value as single)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified floating-point value.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults  
u.setFloatValue("test", 5)  
MsgBox str(u.FloatForKey("test"))
```

4.30.29 setIntegerValue(key as string, value as Integer)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified integer value.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults  
u.setIntegerValue("test", 5)  
MsgBox str(u.integerForKey("test"))
```

4.30.30 setPersistentDomain(domain as dictionary, domainName as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the dictionary for the specified persistent domain.

Notes: domain: The dictionary of keys and values you want to assign to the domain.

domainName: The domain whose keys and values you want to set. This value should be equal to your application's bundle identifier.

When a persistent domain is changed, an NSUserDefaultsDidChangeNotification is posted.

4.30.31 setStringArrayValue(key as string, values() as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified string array.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim t(-1) as string = array("Hello","World")
```

```
u.setStringArrayValue "test", t
```

```
dim a(-1) as string = u.stringArrayForKey("test")
```

```
MsgBox Join(a) // shows "Hello World"
```

4.30.32 setStringValue(key as string, value as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified text string.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
u.setStringValue "test", "Hello √$√/°"
```

```
MsgBox u.stringForKey("test")
```

Notes: setStringValue is for text with encoding and setDataValue is for strings containing binary data.

4.30.33 setURLValue(key as string, value as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key to the specified URL.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
u.setURLValue "test", "http://www.apple.de/"
```

```
MsgBox u.URLForKey("test")
```

Notes: Available in Mac OS X v10.6 and later.

4.30.34 setVariantValue(key as string, value as Variant)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the value of the specified default key in the standard application domain.

Example:

```
dim u as new NSUserDefaultsMBS
u.setVariantValue "test", "Hello World"
MsgBox u.variantForKey("test")
```

Notes: The value parameter can be only property list objects: String, Number, Date, an Array of variant, or Dictionary. For Arrays and Dictionaries, their contents must be property list objects.

Setting a default has no effect on the value returned by the objectForKey: method if the same key exists in a domain that precedes the application domain in the search list.

4.30.35 setVolatileDomain(domain as dictionary, domainName as string)

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Sets the dictionary for the specified volatile domain.

Notes: domain: The dictionary of keys and values you want to assign to the domain.

domainName: The domain whose keys and values you want to set.

This method raises an NSInvalidArgumentException if a volatile domain with the specified name already exists.

4.30.36 standardUserDefaults as NSUserDefaultsMBS

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns the shared defaults object.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

Notes: If the shared defaults object does not exist yet, it is created with a search list containing the names of the following domains, in this order:

NSArgumentDomain, consisting of defaults parsed from the application’s arguments. A domain identified by the application’s bundle identifier

NSGlobalDomain, consisting of defaults meant to be seen by all applications. Separate domains for each of the user’s preferred languages

NSRegistrationDomain, a set of temporary defaults whose values can be set by the application to ensure that searches will always be successful

The defaults are initialized for the current user. Subsequent modifications to the standard search list remain in effect even when this method is invoked again—the search list is guaranteed to be standard only the first time this method is invoked. The shared instance is provided as a convenience—you can create custom instances using the Constructor.

4.30.37 stringArrayForKey(key as string) as string()

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns the array of strings associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
dim t(-1) as string = array("Hello","World")
```

```
u.setStringArrayValue "test", t
```

```
dim a(-1) as string = u.stringArrayForKey("test")
```

```
MsgBox Join(a) // shows "Hello World"
```

Notes: Returns an array with strings. The array is empty if the value for the key is not an array.

4.30.38 synchronize as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Writes any modifications to the persistent domains to disk and updates all unmodified persistent domains to what is on disk.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
// do something

if u.synchronize then
  MsgBox "Saved preferences."
else
  MsgBox "Failed to save preferences."
end if
```

Notes: Returns true if the data was saved successfully to disk, otherwise false.

Because this method is automatically invoked at periodic intervals, use this method only if you cannot wait for the automatic synchronization (for example, if your application is about to exit) or if you want to update the user defaults to what is on disk even though you have not made any changes.

4.30.39 volatileDomainForName(domainName as string) as dictionary

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns the dictionary for the specified volatile domain.

Notes: Returns the dictionary of keys and values belonging to the domain. The keys in the dictionary are names of defaults, and the value corresponding to each key is a property list object (String, Number, Date, Array, or Dictionary).

4.30.40 volatileDomainNames as string()

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: Returns an array of the current volatile domain names.

Notes: You can get the contents of each domain by passing the returned domain names to the volatileDomainForName method.

4.30.41 Properties

4.30.42 boolForKey(key as string) as boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The Boolean value associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
u.boolForKey("test") = true
MsgBox str(u.boolForKey("test"))
```

Notes: If a boolean value is associated with defaultName in the user defaults, that value is returned. Otherwise, false is returned.
(Read and Write computed property)

4.30.43 dataForKey(key as string) as memoryblock

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The data string associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults

u.dataForKey("test")="Hello √§√√Ω"

MsgBox u.dataForKey("test") // shows no umlauts as encoding is away in binary string
```

Notes: Returns the memoryblock or nil if there is no value for this key.

dataForKey is for any string and does not store encoding. stringForKey does store encoding and works only for text strings.
(Read and Write computed property)

4.30.44 dictionaryForKey(key as string) as dictionary

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The dictionary object associated with the specified key.

Example:

```
dim u as new NSUserDefaultsMBS
dim d as new Dictionary

d.Value("First")="Hello"
d.Value("Second")=1234.5678
u.setDictionaryValue "test", d

dim e as Dictionary = u.dictionaryForKey("test")
for each key as Variant in e.keys
```

```
dim value as Variant = e.Value(key)
MsgBox key+": "+value
next
```

Notes: Returns dictionary object associated with the specified key, or nil if the key does not exist or its value is not an dictionary.
(Read and Write computed property)

4.30.45 doubleForKey(key as string) as Double

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The double value associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.doubleForKey("test") = 5
MsgBox str(u.doubleForKey("test"))
```

Notes: Returns the double value associated with the specified key. If the key does not exist, this method returns 0.
Available in Mac OS X v10.5 and later. On Mac OS X 10.4 the plugin will call floatForKey for you to avoid an exception.
(Read and Write computed property)

4.30.46 fileForKey(key as string) as folderitem

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The NSURL instance associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
dim f as FolderItem = SpecialFolder.Desktop
```

```
// write
u.fileForKey("test")=f
```

```
// read
MsgBox u.fileForKey("test").NativePath
```

Notes: Available in Mac OS X v10.6 and later.
(Read and Write computed property)

4.30.47 floatForKey(key as string) as single

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The floating-point value associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.floatForKey("test") = 5
MsgBox str(u.floatForKey("test"))
```

Notes: Returns the floating-point value associated with the specified key. If the key does not exist, this method returns 0.

(Read and Write computed property)

4.30.48 integerForKey(key as string) as Integer

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The integer value associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.integerForKey("test") = 5
MsgBox str(u.integerForKey("test"))
```

Notes: The integer value associated with the specified key. If the specified key does not exist, this method returns 0.

(Read and Write computed property)

4.30.49 stringForKey(key as string) as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The string associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
```

```
u.setStringValue "test", "Hello √$√√°"
MsgBox u.stringForKey("test")
```

Notes: Returns the string or "" if there is no value for this key.

dataForKey is for any string and does not store encoding. stringForKey does store encoding and works only for text strings.

(Read and Write computed property)

4.30.50 URLForKey(key as string) as string

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The NSURL instance associated with the specified key.

Example:

```
dim u as NSUserDefaultsMBS = NSUserDefaultsMBS.standardUserDefaults
u.setURLValue "test", "http://www.apple.de/"
MsgBox u.URLForKey("test")
```

Notes: Available in Mac OS X v10.6 and later.

(Read and Write computed property)

4.30.51 variantForKey(key as string) as Variant

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The object associated with the first occurrence of the specified default.

Example:

```
dim u as new NSUserDefaultsMBS
u.setVariantValue "test", "Hello World"
MsgBox u.variantForKey("test")
```

Notes: This method searches the domains included in the search list in the order they are listed.

(Read and Write computed property)

4.31 class NSUUIDMBS

4.31.1 class NSUUIDMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: An object representing a universally unique value that bridges to UUID; use NSUUID when you need reference semantics or other Foundation-specific behavior.

Notes: UUIDs (Universally Unique Identifiers), also known as GUIDs (Globally Unique Identifiers) or IIDs (Interface Identifiers), are 128-bit values. UUIDs created by NSUUID conform to RFC 4122 version 4 and are created with random bytes.

The standard format for UUIDs represented in ASCII is a string punctuated by hyphens, for example 68753A44-4D6F-1226-9C60-0050E4C00067. The hex representation looks, as you might expect, like a list of numerical values preceded by 0x. For example, 0xD7, 0x36, 0x95, 0x0A, 0x4D, 0x6E, 0x12, 0x26, 0x80, 0x3A, 0x00, 0x50, 0xE4, 0xC0, 0x00, 0x67. Because a UUID is expressed simply as an array of bytes, there are no endianness considerations for different platforms.

Blog Entries

- [MBS Xojo Plugins, version 18.2pr2](#)

4.31.2 Methods

4.31.3 Available as boolean

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true on macOS 10.8 or newer.

4.31.4 Constructor

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Initializes a new UUID with RFC 4122 version 4 random bytes.

See also:

- 4.31.5 Constructor(UUID as MemoryBlock) 419
- 4.31.6 Constructor(UUID as String) 420

4.31.5 Constructor(UUID as MemoryBlock)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Initializes a new UUID with the given bytes.

See also:

- 4.31.4 Constructor 419
- 4.31.6 Constructor(UUID as String) 420

4.31.6 Constructor(UUID as String)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Initializes a new UUID with the formatted string.

See also:

- 4.31.4 Constructor 419
- 4.31.5 Constructor(UUID as MemoryBlock) 419

4.31.7 copy as NSUUIDMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Copies the UUID.

4.31.8 isEqual(other as NSUUIDMBS) as boolean

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Compares two UUIDs.

Example:

```
dim u1 as new NSUUIDMBS
dim s1 as string = u1.UUIDString

// make new object with same UUID
dim u2 as NSUUIDMBS = new NSUUIDMBS(s1)
dim s2 as string = u2.UUIDString

// other uuid
dim o as new NSUUIDMBS
dim so as string = o.UUIDString

dim e1 as Boolean = u1.isEqual(u2)
dim e2 as Boolean = u1.isEqual(o)
```

Break

Notes: Returns true if both are equal.

4.31.9 Operator_Compare(other as NSUUIDMBS) as Integer

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Compares two UUIDs.

Example:

```
dim u1 as new NSUUIDMBS
dim s1 as string = u1.UUIDString

// make new object with same UUID
dim u2 as NSUUIDMBS = new NSUUIDMBS(s1)
dim s2 as string = u2.UUIDString

// other uuid
dim o as new NSUUIDMBS
dim so as string = o.UUIDString

if u1 = u2 then
  MsgBox "u1 and u2 are equal."
else
  MsgBox "u1 and u2 are not equal. Problem?"
end if

if u1 = o then
  MsgBox "u1 and o are equal. Problem?"
else
  MsgBox "u1 and o are not equal."
end if
```

4.31.10 UUID as NSUUIDMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Create and returns a new UUID with RFC 4122 version 4 random bytes.

4.31.11 Properties

4.31.12 data as MemoryBlock

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Queries copy of the bytes as memory block.

Notes: (Read only property)

4.31.13 Handle as Integer

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

4.31.14 UUIDString as String

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: The UUID as a string.

Notes: (Read only property)

4.32 class NSViewControllerMBS

4.32.1 class NSViewControllerMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: The plugin class for a NSViewController.

Notes: Available in Mac OS X v10.5 and later.

Subclass of the NSResponderMBS class.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [MBS Xojo Plugins, version 18.4pr6](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr1](#)
- [MBS Real Studio Plugins, version 11.2pr11](#)
- [MBS Real Studio Plugins, version 11.2pr9](#)

Xojo Developer Magazine

- [12.4, page 31: Writing OS X Apps for Game Center, Getting started by authenticating the local player with the Game Center Sandbox testing facility by Tom Baumgartner](#)

4.32.2 Methods

4.32.3 available as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this class is available.

Example:

```
MsgBox str(NSViewControllerMBS.available)
```

Notes: Returns true on Mac OS X 10.5 or newer.

4.32.4 Constructor

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor creating a new view controller.

4.32.5 `contentViewController(window as NSWindowMBS) as NSViewControllerMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Queries content view's content view controller for the window.

Notes: Returns nil if the window has no controller.

The main content view controller for the window.

The value of this property provides the content view of the window. Setting this value removes the existing value of `contentView` and makes the `contentViewController.view` the main content view for the window. By default, the value of this property is nil.

4.32.6 `dismissViewController(ViewController as NSViewControllerMBS)`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Dismisses a presented view controller, using the same animator that presented it.

Notes: In macOS, this is the universal way to dismiss a view controller, no matter how it was presented.

4.32.7 `loadView`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Explicitly load the view now.

Notes: This method connects an instantiated view from a nib file to the view property of the view controller. This method is called by the system, and is exposed in this class so you can override it to add behavior immediately before or after nib loading.

Do not call this method. If you require this method to be called, access the view property.

Do not invoke this method from other objects unless you take care to avoid redundant invocations. The default implementation of the `loadView` method handles redundant invocations correctly, but a view controller subclass might not. To be safe, other objects should instead access a view controller's `view` property.

The `loadView` method first obtains the values of the view controller's `nibName` and `nibBundle` properties. It then employs the `NSBundle` class to instantiate the specified nib file via the `initWithOwner:topLevelObjects:` method, providing the view controller object as the owner parameter.

For this method to work correctly, you need to have specified the file's owner of the nib file, in Interface Builder, to be `NSViewController`. You also need to have correctly connected the view outlet of the file's owner to the intended view in the nib file. Then, at runtime, the nib loading machinery sets the value of the view controller's `view` property to the nib file's instantiated view.

Prior to OS X v10.10, the `loadView` method did not provide well-defined behavior if the `nibName` property's value was `nil`. In macOS 10.10 and later, however, you get correct behavior without specifying a nib name as long as the nib file's name is the same as that of the view controller. For example, if you have a view controller subclass called `MyViewController` and a nib file with the same name, you can employ the convenient initialization pattern `[[MyViewController alloc] init]`.

4.32.8 `presentViewControllerAsModalWindow(ViewController as NSViewControllerMBS)`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Presents another view controller as a modal window, also known as an alert.

Notes: To dismiss the modal window, call the `dismissViewController:` method on `self` (the presenting view controller).

4.32.9 `presentViewControllerAsPopover(ViewController as NSViewControllerMBS, RelativeToRect as NSRectMBS, positioningView as NSViewMBS, preferredEdge as Integer, behavior as Integer)`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Presents another view controller as a popover.

Notes: `viewController:` The other view controller to present as a popover.

`RelativeToRect:` The content size of the popover.

`positioningView:` The view relative to which the popover should be positioned. Must not be `nil`, or else the view controller raises an `NSInvalidArgumentException` exception.

`preferredEdge:` The edge of `positioningView` that the popover should prefer to be anchored to.

`behavior:` The popover's closing behavior. See the `NSPopoverBehavior` enumeration.

To dismiss the popover, call the `dismissViewController:` method on `self` (the presenting view controller).

4.32.10 `presentViewControllerAsSheet(ViewController as NSViewControllerMBS)`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Presents a new sheet window with the given view controller.

Notes: The presented view controller is the delegate and the content view controller of its sheet.

To dismiss the sheet, call the `dismissViewController:` method on `self` (the presenting view controller).

4.32.11 `windowWithContentViewController(ViewController as NSViewControllerMBS) as NSWindowMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a titled window that contains the specified content view controller.

Notes: `ViewController`: The view controller that provides the main content view for the window. The window's `contentView` property is set to `contentViewController.view`.

Returns a window with the content view controller set to the passed-in view controller object.

This method creates a basic window object that is titled, closable, resizable, and miniaturizable. By default, the window's title is automatically bound to the title of `contentViewController`. You can control the size of the window by using Auto Layout and applying size constraints to the view or its subviews. The initial size of the window is set to the initial size of `contentView` (that is, the size of `contentViewController.view`). The newly created window has `releasedWhenClosed` set to NO, and it must be explicitly retained to keep the window instance alive.

4.32.12 Properties

4.32.13 `className` as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop & iOS.

Function: The name of this `NSViewController` class.

Notes: (Read only property)

4.32.14 `classPath` as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop & iOS.

Function: The path of this `NSViewController` class.

Notes: Useful for debugging to know what super classes the view controller has.
(Read only property)

4.32.15 `Identifier` as String

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

Function: A string that identifies this user interface item.

Notes: It should be set to a unique value on `NSViews` when they are intended to be used inside a view-based `NSTableView`. Identifiers should be unique per-window. For programmatically created user interface items,

you would typically set this value in code after creating a view but before adding it to a window. You may also want to set an identifier on a window, after creating it programmatically, to identify the window easily when it is reopened. You should not change the identifier after a view is added to a window. Identifiers beginning with an underscore are reserved for the system. In framework classes that implement this protocol, the accessor methods are not intended to be overridden.

To help avoid collision of identifiers, it is recommended that identifiers use the same prefix as is used for the framework or application. For example, identifiers for standard AppKit interface items, such as the open panel, will begin with "NS".

The slash '/', backslash '\', and colon ':' characters are reserved and should not be used in identifiers.
(Read and Write property)

4.32.16 representedObject as Variant

Plugin Version: 18.4, Platform: macOS, Targets: Desktop & iOS.

Function: The object whose value is presented in the receiver,Ãs primary view.

Notes: (Read and Write property)

4.32.17 Title as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: The localized title of the receiver's view.

Notes: NSViewController does not use the title property directly. This property is here because so many anticipated uses of this class will involve letting the user choose among multiple named views using a pull-down menu or some other user interface.

(Read and Write property)

4.32.18 view as NSViewMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: The view belonging to this view controller.

Notes: (Read and Write property)

4.32.19 viewLoaded as Boolean

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

Function: Whether view is already loaded.

Notes: (Read only property)

4.32.20 Constants

Edge Constants

Constant	Value	Description
<code>NSMaxXEdge</code>	2	the maximum X edge. Typically right side.
<code>NSMaxYEdge</code>	3	The maximum Y edge. Typically the top edge of a window.
<code>NSMinXEdge</code>	0	the minimum X edge. Typically left side.
<code>NSMinYEdge</code>	1	Minimum Y. As coordinates are upside down in the Cocoa world, this is the bottom edge of a window.

Behavior modes

Constant	Value	Description
<code>PopoverBehaviorApplicationDefined</code>	0	Your application assumes responsibility for closing the popover. AppKit still close the popover in a limited number of circumstances. For instance, AppKit will attempt to close the popover when the window of its positioning view is closed. The exact interactions in which AppKit will close the popover are not guaranteed. You may consider implementing <code>-cancel:</code> to close the popover when the escape key is pressed.
<code>PopoverBehaviorSemitransient</code>	2	AppKit will close the popover when the user interacts with user interface elements in the window containing the popover's positioning view. Semi-transient popovers cannot be shown relative to views in other popovers, nor can they be shown relative to views in child windows. The exact interactions that cause semi-transient popovers to close are not specified.
<code>PopoverBehaviorTransient</code>	1	AppKit will close the popover when the user interacts with a user interface element outside the popover. Note that interacting with menus or panels that become key only when needed will not cause a transient popover to close. The exact interactions that will cause transient popovers to close are not specified.

4.33 class NSViewMBS

4.33.1 class NSViewMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: The Cocoa class for a view.

Notes: To draw into a NSView, use either CustomNSViewMBS class with drawRect event or NSGraphicsMBS class.

Subclass of the NSResponderMBS class.

Blog Entries

- [MBS Xojo Plugins, version 23.4pr5](#)
- [News from the MBS Xojo Plugins Version 22.2](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.2](#)
- [News from the MBS Xojo Plugins Version 22.1](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.4](#)
- [MBS Real Studio Plugins, version 12.1fc](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Plugins 10.3 Release Notes](#)
- [MBS REALbasic plug-ins version 9.4](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.6](#)

Xojo Developer Magazine

- [19.6, page 10: News](#)

4.33.2 Methods

4.33.3 addSubview(subview as NSViewMBS)

Plugin Version: 8.0, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a view to the subviews so it's displayed above its siblings.

Example:

```
// create a button  
dim n as new NSButtonMBS(100, 100, 50, 24)
```

```
n.bezelStyle = n.NSRoundedBezelStyle  
n.Title = "Test"
```

```
// add directly to window's content view
window1.NSWindowMBS.contentView.addSubview n
```

See also:

- 4.33.4 addSubview(subview as NSViewMBS, positioned as Integer, relativeToView as NSViewMBS)
430

4.33.4 addSubview(subview as NSViewMBS, positioned as Integer, relativeToView as NSViewMBS)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Inserts a view among the receiver's subviews so it's displayed immediately above or below another view.

Example:

```
// create a button
dim n as new NSButtonMBS(100, 100, 50, 24)
```

```
n.bezelStyle = n.NSRoundedBezelStyle
n.Title = "Test"
```

```
// add directly to window's super view, below the content view
// so it is behind all RB controls
window1.NSWindowMBS.contentView.superview.addSubview n, NSWindowMBS.NSWindowBelow, window1.NSWindowMBS.contentView
```

Notes: subView: The view object to add to the receiver as a subview.

positioned: A constant specifying the position of the aView relative to otherView. Valid values are NSWindowAbove or NSWindowBelow.

relativeToView: The other view subView is to be positioned relative to. If relativeToView is nil (or isn't a subview of the receiver), subView is added above or below all of its new siblings.

This method also sets the receiver as the next responder of aView.

See also:

- 4.33.3 addSubview(subview as NSViewMBS) 429

4.33.5 addToolTipRect(rect as NSRectMBS, tooltip as NSViewToolTipMBS)

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: Creates a tooltip for a defined area the view and returns a tag that identifies the tooltip rectangle.

Notes: Rect: A rectangle defining the region of the view to associate the tooltip with.

tooltip: The object to provide text.

Please keep reference of tooltip object as the tooltip is removed by it's destructor.

4.33.6 ancestorSharedWithView(view as NSViewMBS) as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The closest ancestor shared by the receiver and a given view.

Notes: The closest ancestor or nil if there's no such object. Returns self if aView is identical to the receiver.

4.33.7 animator as NSViewMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop only.

Function: Returns the proxy object for this view which animates.

Example:

```
dim v as NSViewMBS // your NSViewMBS object
```

```
v.alphaValue = 0.5 // switch alpha directly
```

```
v.animator.alphaValue = 0.5 // switch alpha animated
```

4.33.8 backgroundFilters as variant()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: An array of Core Image filters to apply to the view,Ãs background.

Notes: This property contains an array of CIFilterMBS objects. This array represents the background filters stored in the backgroundFilters property of the view,Ãs layer. If the view does not have a layer, setting the value of this property has no effect.

The default value of this property is an empty array.

We use variant here to reduce plugin dependencies. Please use with array of variants containing CIFilterMBS.

4.33.9 beginDraggingSessionWithItems(items() as Variant, e as NSEventMBS, source as NSViewMBS) as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Initiates a dragging session with a group of dragging items.

Notes: items: The dragging items (array of variant containing NSDraggingItemMBS objects). The frame property of each NSDraggingItem must be in the view's coordinate system.

event: The mouse-down event object from which to initiate the drag operation. In particular, its mouse location is used for the offset of the icon being dragged.

source: An object that serves as the controller of the dragging operation. It must conform to the NSDraggingSource informal protocol and is typically the receiver itself or its NSWindow object.

Returns the dragging session for the drag (NSDraggingSessionMBS object). Returned as Variant to reduce plugin dependencies.

A basic drag starts by calling beginDraggingSessionWithItems.

The caller can take the returned NSDraggingSession and continue to modify its properties such as slidesBackOnCancelOrFail. When the drag actually starts, the source is sent a draggingSessionWillBeginAtPoint message followed by multiple draggingSessionMovedToPoint messages as the user drags.

Once the drag is ended or cancelled, the source receives a draggingSessionEndedAtPoint:operation and the drag is complete.

Available in OS X v10.7 and later.

4.33.10 Constructor

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Creates a new control with size 100/100 and position 0/0

Notes: On success the handle property is not zero.

See also:

- 4.33.11 Constructor(Handle as Integer) 432
- 4.33.12 Constructor(left as Double, top as Double, width as Double, height as Double) 433

4.33.11 Constructor(Handle as Integer)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop & iOS.

Function: Creates an object based on the given NSView handle.

Example:

```
dim t as new NSTextViewMBS(0, 0, 100, 100)
dim v as new NSTextViewMBS(t.handle)
```

```
MsgBox str(v.Bounds.Width)+" x "+str(v.Bounds.Height)
```

Notes: The handle is casted to a NSView and the plugin retains this handle.

See also:

- 4.33.10 Constructor 432
- 4.33.12 Constructor(left as Double, top as Double, width as Double, height as Double) 433

4.33.12 Constructor(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 7.8, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor to create a new NSView.

Notes: On success handle is set.

For a toolbar, please get the view directly: self.NSToolbarMBS.toolbarView.

See also:

- 4.33.10 Constructor 432
- 4.33.11 Constructor(Handle as Integer) 432

4.33.13 contentFilters as variant()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: An array of Core Image filters to apply to the contents of the view and its sublayers.

Notes: This property contains an array of CIFilterMBS objects. This array represents the filters stored in the filters property of the view,Äs layer. If the view does not have a layer, setting the value of this property has no effect.

The default value of this property is an empty array.

We use variant here to reduce plugin dependencies. Please use with array of variants containing CIFilterMBS.

4.33.14 `convertPointFromView(point as NSPointMBS, View as NSViewMBS) as NSPointMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a point from the coordinate system of a given view to that of the receiver.

Notes: point: A point specifying a location in the coordinate system of aView.

View: The view with aPoint in its coordinate system. If View is nil, this method instead converts from window base coordinates. Otherwise, both View and the receiver must belong to the same NSWindow object.

Returns the point converted to the coordinate system of the receiver.

4.33.15 `convertPointToView(point as NSPointMBS, View as NSViewMBS) as NSPointMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a point from the receiver's coordinate system to that of a given view.

Notes: point: A point specifying a location in the coordinate system of the receiver.

View: The view into whose coordinate system point is to be converted. If View is nil, this method instead converts to window base coordinates. Otherwise, both View and the receiver must belong to the same NSWindow object.

Returns the point converted to the coordinate system of View.

4.33.16 `convertRectFromView(rect as NSRectMBS, View as NSViewMBS) as NSRectMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a rectangle from the coordinate system of another view to that of the receiver.

Notes: rect: The rectangle in aView's coordinate system.

view: The view with aRect in its coordinate system. If view is nil, this method instead converts from window base coordinates. Otherwise, both view and the receiver must belong to the same NSWindow object.

Returns the converted rectangle.

4.33.17 `convertRectToView(rect as NSRectMBS, View as NSViewMBS) as NSRectMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a rectangle from the receiver's coordinate system to that of another view.

Notes: rect: A rectangle in the receiver's coordinate system.

View: The view that is the target of the conversion operation. If View is nil, this method instead converts from window base coordinates. Otherwise, both View and the receiver must belong to the same NSWindow object.

Returns the converted rectangle.

4.33.18 `convertSizeFromView(Size as NSSizeMBS, View as NSViewMBS) as NSSizeMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a size from another view's coordinate system to that of the receiver.

Notes: Size: The size (width and height) in aView's coordinate system.

View: The view with Size in its coordinate system. If View is nil, this method instead converts from window base coordinates. Otherwise, both View and the receiver must belong to the same NSWindow object.

Returns the converted size, as an NSSize structure.

4.33.19 `convertSizeToView(Size as NSSizeMBS, View as NSViewMBS) as NSSizeMBS`

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: Converts a size from the receiver's coordinate system to that of another view.

Notes: Size: The size (width and height) in the receiver's coordinate system.

View: The view that is the target of the conversion operation. If View is nil, this method instead converts from window base coordinates. Otherwise, both View and the receiver must belong to the same NSWindow object.

Returns the converted size, as an NSSize structure.

4.33.20 `dataWithEPSInsideRect(left as Double, top as Double, width as Double, height as Double) as Memoryblock`

Plugin Version: 8.5, Platform: macOS, Targets: Desktop only.

Function: Returns EPS data that draws the region of the receiver within a specified rectangle.

Example:

```

dim n as new NSProgressIndicatorMBS
n.sizeToFit

dim data as string = n.dataWithEPSInsideRect(0, 0, n.frameWidth, n.frameHeight)
dim f as FolderItem = SpecialFolder.Desktop.Child("test.ps")
dim b as BinaryStream = f.CreateBinaryFile("")

b.Write data
f.Launch

```

4.33.21 dataWithPDFInsideRect(left as Double, top as Double, width as Double, height as Double) as Memoryblock

Plugin Version: 8.5, Platform: macOS, Targets: Desktop only.

Function: Returns PDF data that draws the region of the receiver within a specified rectangle.

Example:

```

dim n as new NSProgressIndicatorMBS
n.sizeToFit

dim data as string = n.dataWithPDFInsideRect(0, 0, n.frameWidth, n.frameHeight)
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
dim b as BinaryStream = f.CreateBinaryFile("")

b.Write data
f.Launch

```

See also:

- 4.33.22 dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock 436

4.33.22 dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Returns PDF data that draws the region of the receiver within a specified rectangle.

See also:

- 4.33.21 dataWithPDFInsideRect(left as Double, top as Double, width as Double, height as Double) as Memoryblock 436

4.33.23 dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NSSizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Initiates a dragging operation from the receiver, allowing the user to drag arbitrary data with a specified icon into any application that has window or view objects that accept dragged data.

Notes: Image: The NSImage object to be dragged.

viewLocation: The location of the image's lower-left corner, in the receiver's coordinate system. It determines the placement of the dragged image under the cursor. When determining the image location you should use the mouse down coordinate, provided in theEvent, rather than the current mouse location.

offset: This parameter is ignored.

NSEvent: The left mouse-down event that triggered the dragging operation (see discussion below).

pboard: The pasteboard that holds the data to be transferred to the destination (see discussion below).

source: An object that serves as the controller of the dragging operation. It must conform to the NSDraggingSource protocol and is typically the receiver itself or its NSWindow object.

slideBack: A Boolean that determines whether the drag image should slide back if it's rejected. The image slides back to imageLoc if slideBack is true and the image isn't accepted by the dragging destination. If false the image doesn't slide back.

This method must be invoked only within an implementation of the mouseDown or mouseDragged methods. Before invoking this method, you must place the data to be transferred on pboard. To do this, get the drag pasteboard object (NSDragPboard), declare the types of the data, and then put the data on the pasteboard.

4.33.24 drawFocusRingMask

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Draws the focus ring mask for the view.

Notes: This method provides the shape of the focus ring mask by drawing the focus ring mask. An implementation of this method should draw in the view's interior (bounds) coordinate space, that the focus ring style has been set (it will be set it to NSFocusingOnly to capture the focus ring itself), and that the fill and stroke colors have been set to an arbitrary fully opaque color.

Subclasses that find the default behavior insufficient should only draw the focus ring shape.

The NSView implementation of this method simply fills self.bounds.
Available in Mac OS X v10.7 and later.

4.33.25 enclosingMenuItem as Variant

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Returns the menu item containing the receiver or any of its superviews in the view hierarchy.

Notes: Returns the menu item containing the receiver or any of its superviews in the view hierarchy, or nil if the receiver's view hierarchy is not in a menu item

Available in Mac OS X v10.5 and later.

Declared as a variant to reduce plugin dependencies.

Please assign to a NSScrollViewMBS variable.

4.33.26 enclosingScrollView as Variant

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the nearest ancestor NSScrollView object containing the receiver (not including the receiver itself); otherwise returns nil.

Notes: Declared as a variant to reduce plugin dependencies.

Please assign to a NSScrollViewMBS variable.

4.33.27 focusRingMaskBounds as NSRectMBS

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Returns the focus ring mask bounds.

Notes: Returns a rectangle containing the mask in the view's interior (bounds) coordinate space.

The mask bounds allows the focus ring's overall size and position to be determined before it is drawn.

Subclasses must override this method if they require the display of a focus ring.

The NSView implementation of this method simply returns NSRectMBS.Zero.

Note: The information provided by focusRingMaskBounds will enable Accessibility to identify selected subelements for zoom tracking, so it is important that this method provide a reasonably tight bounding box and that noteFocusRingMaskChanged is invoked as described.

4.33.28 isDescendantOf(view as NSViewMBS) as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if the receiver is a subview of a given view or if it's identical to that view; otherwise, it returns false.

4.33.29 makeBackingLayer as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Creates the view's backing layer.

Notes: Value is a CALayerMBS object.

Available in OS X v10.6 and later.

4.33.30 nextValidKeyView as NSViewMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns the closest view object in the key view loop that follows the receiver and accepts first responder status.

Notes: This method ignores hidden views when it determines the next valid key view.

Returns nil on any error.

4.33.31 noteFocusRingMaskChanged

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Invoked to notify the view that the focus ring mask requires updating.

Notes: It is important to note that it is only necessary for developers to invoke this method when some internal state change of their application, that the Application Kit can't determine, affects the shape of the focus ring mask.

It is assumed that if the view is marked as needing display, or is resized, its focus ring shape is likely to have changed, and there is no need for clients to explicitly send this message in such cases, they are handled automatically.

If, however, a view is showing a focus ring around some part of its content (an NSImage, perhaps), and that content changes, the client must provide notification by invoking this method so that focusRingMaskBounds and drawFocusRingMask will be invoked to redraw the focus ring.

Available in Mac OS X v10.7 and later.

4.33.32 `NSViewBoundsDidChangeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to use with the `NSView` class.

Notes: Posted whenever the `NSView`'s bounds rectangle changes independently of the frame rectangle, if the `NSView` is configured using `setPostsBoundsChangedNotifications` to post such notifications.

The notification object is the `NSView` object whose bounds rectangle has changed. This notification does not contain a `userInfo` dictionary.

The following methods can result in notification posting

- `setBounds`
- `setBoundsOrigin`
- `setBoundsRotation`
- `setBoundsSize`
- `translateOriginToPoint`
- `scaleUnitSquareToSize`
- `rotateByAngle`

Note that the bounds rectangle resizes automatically to track the frame rectangle. Because the primary change is that of the frame rectangle, however, `setFrame` and `setFrameSize` don't result in a bounds-changed notification.

4.33.33 `NSViewDidUpdateTrackingAreasNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to use with the `NSView` class.

Notes: Posted whenever an `NSView` object recalculates its tracking areas. It is sent after the view receives `updateTrackingAreas`.

Available in Mac OS X v10.5 and later.

4.33.34 `NSViewFocusDidChangeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to use with the `NSView` class.

Notes: Sent when focus changed. Object for this notification is the view.

4.33.35 `NSViewFrameDidChangeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to use with the `NSView` class.

Notes: Posted whenever the view's frame rectangle changes, if the view is configured using `setPostsFrameChangedNotifications` to post such notifications.

The notification object is the `NSView` object whose frame rectangle has changed. This notification does not contain a `userInfo` dictionary.

The following methods can result in notification posting:

- `setFrame`
- `setFrameOrigin`
- `setFrameRotation`
- `setFrameSize`

4.33.36 `NSViewGlobalFrameDidChangeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop only.

Function: One of the notification names to use with the `NSView` class.

Notes: Posted whenever an `NSView` object that has attached surfaces (that is, `NSOpenGLContext` objects) moves to a different screen, or other cases where the `NSOpenGLContext` object needs to be updated. The notification object is the surface's view. This notification does not contain a `userInfo` dictionary.

4.33.37 `pageFooter` as `NSAttributedStringMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns a default footer string that includes the current page number and page count.

Notes: A printable view class can override this method to substitute its own content in place of the default value. You should not need to call this method directly. The printing system calls it once per page during printing.

Footers are generated only if the user defaults contain the key `NSPrintHeaderAndFooter` with the value `true`.

4.33.38 `pageHeader` as `NSAttributedStringMBS`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: Returns a default header string that includes the print job title and date.

Notes: Typically, the print job title is the same as the window title. A printable view class can override this method to substitute its own content in place of the default value. You should not need to call this method directly. The printing system calls it once per page during printing.

Headers are generated only if the user defaults contain the key `NSPrintHeaderAndFooter` with the value `true`.

4.33.39 `previousKeyView` as `NSViewMBS`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns the view object preceding the receiver in the key view loop.

Notes: This view should, if possible, be made first responder when the user navigates backward from the receiver using keyboard interface control.

Returns nil on any error.

4.33.40 `previousValidKeyView` as `NSViewMBS`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns the closest view object in the key view loop that precedes the receiver and accepts first responder status.

Notes: This method ignores hidden views when it determines the previous valid key view.

Returns nil on any error.

4.33.41 `print`

Plugin Version: 12.4, Platform: macOS, Targets: Desktop only.

Function: This action method opens the Print panel, and if the user chooses an option other than canceling, prints the receiver and all its subviews to the device specified in the Print panel.

4.33.42 registeredDraggedTypes as string()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the array of pasteboard drag types that the view can accept.

Example:

```
// we register some types
ListBox1.AcceptPictureDrop
ListBox1.AcceptTextDrop

// and query them:
MsgBox Join(Listbox1.NSViewMBS.registeredDraggedTypes, EndOfLine)

' shows:
'NeXT TIFF v4.0 pasteboard type
'com.apple.traditional-mac-plain-text
'NSStringPboardType
```

Notes: This method returns the types registered by calling `registerForDraggedTypes`. Each element of the array is a uniform type identifier. The returned elements are in no particular order, but the array is guaranteed not to contain duplicate entries.

4.33.43 registerForDraggedTypes(Types() as string)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Registers the pasteboard types that the receiver will accept as the destination of an image-dragging session.

Notes: Types: An array of uniform type identifiers. See `Types for Standard Data` (Mac OS X 10.6 and later) for descriptions of the pasteboard type identifiers.

Registering an `NSView` object for dragged types automatically makes it a candidate destination object for a dragging session. As such, it must properly implement some or all of the `NSDraggingDestination` protocol methods. As a convenience, `NSView` provides default implementations of these methods. See the `NSDraggingDestination` protocol specification for details.

Xojo with Cocoa target implements the methods listed above. So this method is useful to change the allowed types for the case the framework has a bug.

4.33.44 removeAllToolTips

Plugin Version: 11.0, Platform: macOS, Targets: Desktop only.

Function: Removes all tool tips assigned to the receiver.

4.33.45 removeFromSuperview

Plugin Version: 8.0, Platform: macOS, Targets: Desktop & iOS.

Function: Unlinks the view from its superview and its window, removes it from the responder chain, and invalidates its cursor rectangles.

4.33.46 removeFromSuperviewWithoutNeedingDisplay

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Unlinks the receiver from its superview and its window and removes it from the responder chain, but does not invalidate its cursor rectangles to cause redrawing.

Notes: The receiver is also released; if you plan to reuse it, be sure to retain it before sending this message and to release it as appropriate when adding it as a subview of another view.

Unlike its counterpart, `removeFromSuperview`, this method can be safely invoked during display.

4.33.47 RenderImage(subviews as boolean = false, flipped as boolean = false) as variant

Plugin Version: 8.5, Platform: macOS, Targets: Desktop only.

Function: Renders a picture of the view.

Example:

```
dim n as new NSProgressIndicatorMBS
n.sizeToFit
```

```
dim image as NSImageMBS = n.RenderImage
dim pic as Picture = image.CopyPicture
```

```
Backdrop = pic
```

Notes: May return nil on any error.

The value of this variant must be an object of class NSImageMBS.

Added flipped parameter for version 20.3.

Flipped was true for older versions.

4.33.48 `replaceSubview(oldView as NSViewMBS, newView as NSViewMBS)`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop only.

Function: Replaces one of the receiver's subviews with another view.

Notes: `oldView`: The view to be replaced by `newView`. May not be nil.

`newView`: The view to replace `oldView`. May not be nil.

This method does nothing if `oldView` is not a subview of the receiver.

Neither `oldView` nor `newView` may be nil, and the behavior is undefined if either of these parameters is nil.

4.33.49 `rotateByAngle(angle as Double)`

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Rotates the receiver's bounds rectangle by a specified degree value around the origin of the coordinate system, (0.0, 0.0).

Notes: `angle`: A float value specifying the angle of rotation, in degrees.

See the `BoundsRotation` property description for more information. This method neither redisplay the receiver nor marks it as needing display. You must do this yourself with `display` or `NeedsDisplay`.

This method posts an `NSViewBoundsDidChangeNotification` to the default notification center if the receiver is configured to do so.

4.33.50 `scaleUnitSquareToSize(size as NSSizeMBS)`

Plugin Version: 11.0, Platform: macOS, Targets: Desktop only.

Function: Scales the receiver's coordinate system so that the unit square scales to the specified dimensions.

Notes: For example, a `newUnitSize` of (0.5, 1.0) causes the receiver's horizontal coordinates to be halved, in turn doubling the width of its bounds rectangle. Note that scaling is performed from the origin of the coordinate system, (0.0, 0.0), not the origin of the bounds rectangle; as a result, both the origin and size of the bounds rectangle are changed. The frame rectangle remains unchanged.

This method neither redisplay the receiver nor marks it as needing display. You must do this yourself with `display` or `setNeedsDisplay`.

This method posts an `NSNotification` to the default notification center if the receiver is configured to do so.

4.33.51 Screenshot as Picture

Plugin Version: 22.2, Platform: macOS, Targets: Desktop only.

Function: Renders the view into a picture.

Notes: Like `ScreenshotWindowMBS`, but only for the view you are interested in, e.g. a map. Returns nil on failure like out of memory.

4.33.52 setBackgroundFilters(filters() as variant)

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: Sets an array of Core Image filters to apply to the view's background.

Notes: This property contains an array of `CIFilterMBS` objects. This array represents the background filters stored in the `backgroundFilters` property of the view's layer. If the view does not have a layer, setting the value of this property has no effect.

The default value of this property is an empty array.

We use variant here to reduce plugin dependencies. Please use with array of variants containing `CIFilterMBS`.

4.33.53 setBoundsOrigin(origin as NSPointMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the bounds origin.

See also:

- 4.33.54 `setBoundsOrigin(x as Double, y as Double)`

446

4.33.54 setBoundsOrigin(x as Double, y as Double)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the bounds origin.

See also:

- 4.33. CLASS *NSVIEWMBS* 447
- 4.33.53 `setBoundsOrigin(origin as NSPointMBS)` 446

4.33.55 `setBoundsSize(size as NSSizeMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the bounds size of the `nview`.

See also:

- 4.33.56 `setBoundsSize(width as Double, height as Double)` 447

4.33.56 `setBoundsSize(width as Double, height as Double)`

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the bounds size of the `nview`.

See also:

- 4.33.55 `setBoundsSize(size as NSSizeMBS)` 447

4.33.57 `setContentFilters(filters() as variant)`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: Sets an array of Core Image filters to apply to the contents of the view and its sublayers.

Notes: This property contains an array of `CIFilterMBS` objects. This array represents the filters stored in the `filters` property of the view's layer. If the view does not have a layer, setting the value of this property has no effect.

The default value of this property is an empty array.

We use `variant` here to reduce plugin dependencies. Please use with array of variants containing `CIFilterMBS`.

4.33.58 `setFocus`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Convenience function to set the focus to this view.

Notes: Calls internally `makeFirstResponder` on `NSWindow`.

4.33.59 `setFrameOrigin(origin as NSPointMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Sets the frame origin.

See also:

- 4.33.60 setFrameOrigin(x as Double, y as Double) 448

4.33.60 setFrameOrigin(x as Double, y as Double)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Sets the frame origin.

See also:

- 4.33.59 setFrameOrigin(origin as NSPointMBS) 447

4.33.61 setFrameSize(size as NSSizeMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Sets the frame size.

See also:

- 4.33.62 setFrameSize(width as Double, height as Double) 448

4.33.62 setFrameSize(width as Double, height as Double)

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Sets the frame size.

See also:

- 4.33.61 setFrameSize(size as NSSizeMBS) 448

4.33.63 subviews(recursive as boolean = false) as NSViewMBS()

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The immediate subviews.

Notes: The order of the subviews may be considered as being back-to-front, but this does not imply invalidation and drawing behavior. The order is based on the order of the receiver's subviews as specified in the nib file from which they were unarchived or the programmatic interface for modifying the receiver's subview list. This ordering is also the reverse of the order in which hit-testing is done.

If recursive is true, we include all subviews of all subviews. (new in 14.2)

4.33.64 unregisterDraggedTypes

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Unregisters the receiver as a possible destination in a dragging session.

4.33.65 Properties

4.33.66 acceptsTouchEvents as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the view will accept touch events.

Notes: Available in Mac OS X v10.6 and later.

(Read and Write property)

4.33.67 allowsVibrancy as Boolean

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

Function: allowsVibrancy is queried when a vibrant appearance is used on a view hierarchy.

Notes: When allowsVibrancy returns YES, the view will have an appropriate measure taken to ensure it is vibrant on top of its given material.

Specific subclasses, such as NSControl, will answer this question based on the artwork they draw for a given appearance.

(Read only property)

4.33.68 alphaValue as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: the opacity of the view.

Example:

```
// makes a checkbox half transparent on Cocoa target:
```

```
CheckBox1.NSViewMBS.alphaValue = 0.5
```

Notes: This method returns the value of the opacity property of the view's layer. Possible values are between 0.0 (transparent) and 1.0 (opaque). The default is 1.0.

Sending this message to a view that is not managing a Core Animation layer causes an exception.

(Read and Write property)

4.33.69 `autoresizesSubviews` as `boolean`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: True if the receiver automatically resizes its subviews using `resizeSubviewsWithOldSize` whenever its frame size changes, false otherwise.

Notes: (Read and Write property)

4.33.70 `autoresizingMask` as `Integer`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The receiver's autoresizing mask, which determines how it's resized by the `resizeWithOldSuperviewSize` method.

Notes: If the autoresizing mask is equal to `NSViewNotSizable` (that is, if none of the options are set), then the receiver doesn't resize at all in `resizeWithOldSuperviewSize`.

<code>NSViewNotSizable = 0</code>	The receiver cannot be resized.
<code>NSViewMinXMargin = 1</code>	The left margin between the receiver and its superview is flexible.
<code>NSViewWidthSizable = 2</code>	The receiver's width is flexible.
<code>NSViewMaxXMargin = 4</code>	The right margin between the receiver and its superview is flexible.
<code>NSViewMinYMargin = 8</code>	The bottom margin between the receiver and its superview is flexible.
<code>NSViewHeightSizable = 16</code>	The receiver's height is flexible.
<code>NSViewMaxYMargin = 32</code>	The top margin between the receiver and its superview is flexible.

(Read and Write property)

4.33.71 `bounds` as `NSRectMBS`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the receiver's bounds rectangle, which expresses its location and size in its own coordinate system.

Example:

```
dim n as new NSProgressIndicatorMBS
n.sizeToFit
```

```
MsgBox n.bounds.String
```

Notes: The bounds rectangle may be rotated; use the `boundsRotation` method to check this.
(Read and Write property)

4.33.72 `boundsRotation` as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The angle, in degrees, of the view's bounds rectangle relative to its frame rectangle.

Notes: (Read and Write property)

4.33.73 `canBecomeKeyView` as boolean

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Returns whether the receiver can become key view.

Notes: Returns true if the receiver can become key view, false otherwise.

(Read only property)

4.33.74 `canDraw` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns true if drawing commands will produce any result, false otherwise.

Notes: Use this method when invoking a draw method directly along with `lockFocus` and `unlockFocus`, bypassing the `display...` methods (which test drawing ability and perform locking for you). If this method returns false, you shouldn't invoke `lockFocus` or perform any drawing.

A view object can draw on-screen if it is not hidden, it is attached to a view hierarchy in a window (NSWindow), and the window has a corresponding window device. A view object can draw during printing if it is a descendant of the view being printed.

To draw into a NSView, use either CustomNSViewMBS class with `drawRect` event or NSGraphicsMBS class.
(Read only property)

4.33.75 `canDrawConcurrently` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the view's `drawRect:` method can be invoked on a background thread.

Notes: Available in Mac OS X v10.6 and later.

(Read and Write property)

4.33.76 `className` as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop & iOS.

Function: The name of this `NSView` class.

Example:

```
// shows RBNSButton
MsgBox checkbox1.NSViewMBS.className
```

Notes: (Read only property)

4.33.77 `classPath` as string

Plugin Version: 10.0, Platform: macOS, Targets: Desktop & iOS.

Function: The path of this `NSView` class.

Example:

```
MsgBox TextArea1.NSViewMBS.classPath
// shows "RBNSScrollView:NSScrollView:NSView:NSResponder:NSObject"
```

Notes: Useful for debugging to know what super classes the view has.

(Read only property)

4.33.78 `clipsToBounds` as Boolean

Plugin Version: 23.4, Platforms: macOS, iOS, Targets: Desktop & iOS.

Function: A Boolean value that determines whether subviews are confined to the bounds of the view.

Notes: Setting this value to true causes subviews to be clipped to the bounds of the view. If set to false, subviews whose frames extend beyond the visible bounds of the view aren't clipped.

The default value is false. Some subclasses of `UIView`, like `UIScrollView`, override the default value to true.

(Read and Write property)

4.33.79 focusRingType as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The type of focus ring to be drawn around the receiver.

Notes: This method only sets the desired focus ring type and does not cause the view to draw the actual focus ring. You are responsible for drawing the focus ring in your view's drawRect method whenever your view is made the first responder.

possible values:

NSFocusRingTypeDefault	0
NSFocusRingTypeNone	1
NSFocusRingTypeExterior	2

(Read and Write property)

4.33.80 frame as NSRectMBS

Plugin Version: 9.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the frame size and position of the view.

Example:

```
dim n as new NSProgressIndicatorMBS
n.sizeToFit
```

```
MsgBox n.frame.String
```

Notes: (Read and Write property)

4.33.81 frameCenterRotation as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The receiver's rotation about the layer's position.

Notes: If the application has altered the layer's anchorPoint property, the behavior is undefined. Sending this message to a view that is not managing a Core Animation layer causes an exception.

Available in Mac OS X v10.5 and later.

(Read and Write property)

4.33.82 `frameHeight` as Double

Plugin Version: 9.7, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the view frame.

Example:

```
dim n as new NSProgressIndicatorMBS
MsgBox str(n.frameHeight)
```

Notes: This is a convenience property which calls the frame function to get the current rectangle, changes the value and sets the frame to the new rectangle.
(Read and Write property)

4.33.83 `frameLeft` as Double

Plugin Version: 9.7, Platform: macOS, Targets: Desktop & iOS.

Function: The left position of the view frame.

Example:

```
dim n as new NSProgressIndicatorMBS
MsgBox str(n.frameLeft)
```

Notes: This is a convenience property which calls the frame function to get the current rectangle, changes the value and sets the frame to the new rectangle.
(Read and Write property)

4.33.84 `frameRotation` as Double

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The angle, in degrees, of the view's frame relative to its superview's coordinate system.

Notes: dim n as new NSProgressIndicatorMBS

```
n.frameRotation=10
```

```
MsgBox str(n.frameRotation)
```

(Read and Write property)

4.33.85 frameTop as Double

Plugin Version: 9.7, Platform: macOS, Targets: Desktop & iOS.

Function: The top position of the view frame.

Example:

```
dim n as new NSProgressIndicatorMBS
MsgBox str(n.frameTop)
```

Notes: This is a convenience property which calls the frame function to get the current rectangle, changes the value and sets the frame to the new rectangle.

(Read and Write property)

4.33.86 frameWidth as Double

Plugin Version: 9.7, Platform: macOS, Targets: Desktop & iOS.

Function: The width of the view frame.

Example:

```
dim n as new NSProgressIndicatorMBS
MsgBox str(n.frameWidth)
```

Notes: This is a convenience property which calls the frame function to get the current rectangle, changes the value and sets the frame to the new rectangle.

(Read and Write property)

4.33.87 identifier as string

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: A string that identifies this user interface item.

Notes: It should be set to a unique value on NSViews when they are intended to be used inside a view-based NSTableView. Identifiers should be unique per-window. For programmatically created user interface items, you would typically set this value in code after creating a control but before adding it to a window. You may also want to set an identifier on a window, after creating it programmatically, to identify the window easily when it is reopened. You should not change the identifier after a control is added to a window. Identifiers beginning with an underscore are reserved for the system. In framework classes that implement this protocol, the accessor methods are not intended to be overridden.

To help avoid collision of identifiers, it is recommended that identifiers use the same prefix as is used for the

framework or application. For example, identifiers for standard AppKit interface items, such as the open panel, will begin with "NS".

The slash '/', backslash '\', and colon ':' characters are reserved and should not be used in identifiers. (Read and Write property)

4.33.88 isFlipped as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if the nsview uses flipped drawing coordinates or false if it uses native coordinates.

Notes: (Read only property)

4.33.89 isHidden as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver is marked as hidden.

Notes: (Read and Write property)

4.33.90 isHiddenOrHasHiddenAncestor as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if the nsview is marked as hidden or has an ancestor in the view hierarchy that is marked as hidden; returns false otherwise.

Notes: (Read only property)

4.33.91 isOpaque as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Overridden by subclasses to return true if the view is opaque, false otherwise.

Notes: (Read only property)

4.33.92 isRotatedFromBase as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if the `nsview` or any of its ancestors has ever set a `FrameRotation` or `BoundsRotation` properties; otherwise returns false.

Notes: (Read only property)

4.33.93 `isRotatedOrScaledFromBase` as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: True if the `nsview` or any of its ancestors has ever had a nonzero frame or bounds rotation, or has been scaled from the window's base coordinate system; otherwise returns false.

Notes: The intent of this information is to optimize drawing and coordinate calculation, not necessarily to reflect the exact state of the receiver's coordinate system, so it may not reflect the actual rotation or scaling. For example, if an `NSView` object is rotated to 45 degrees and later back to 0, this method still returns true. (Read only property)

4.33.94 `layer` as Variant

Plugin Version: 13.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns or sets the Core Animation layer that the receiver uses as its backing store.

Notes: Value is a `CALayerMBS` object.

Available in OS X v10.5 and later.

(Read and Write property)

4.33.95 `layerUsesCoreImageFilters` as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: A Boolean value indicating whether the view's layer uses Core Image filters and needs in-process rendering.

Notes: If your view uses a custom layer and you assigned Core Image to that layer directly, you must set this property to YES to let AppKit know of that fact. In macOS 10.9 and later, AppKit prefers to render layer trees out-of-process but cannot do so if any layers have Core Image filters attached to them. Specifying true for property lets AppKit know that it must move rendering of the layer hierarchy back into your app's process. If the value of this property is false, adding a filter to the view's layer triggers an exception.

You do not need to modify this property if you assigned the filters using the `backgroundFilters`, `compositingFilter`, or `contentFilters` properties of the view. Those methods automatically let AppKit know that it needs to render the layer hierarchy in-process. Set it only if you set the filters on the layer directly.

(Read and Write property)

4.33.96 needsDisplay as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: whether the view needs to be displayed.

Notes: (Read and Write property)

4.33.97 nextKeyView as NSViewMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: The view object following the receiver in the key view loop.

Notes: Returns the view object following the receiver in the key view loop, or nil if there is none.

On setting inserts a specified view object after the receiver in the key view loop of the receiver's window.

This view should, if possible, be made first responder when the user navigates forward from the receiver using keyboard interface control.

Returns nil on any error.

(Read and Write property)

4.33.98 opaqueAncestor as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the receiver's closest opaque ancestor (including the receiver itself).

Notes: (Read only property)

4.33.99 RetainCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop & iOS.

Function: The retain count for the NSView object.

Notes: Useful for debugging. Should always be >0.

(Read only property)

4.33.100 superview as NSViewMBS

Plugin Version: 7.7, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the receiver's superview, or nil if it has none.

Notes: (Read only property)

4.33.101 tooltip as string

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: The tooltip text for this view.

Example:

```
dim n as new NSProgressIndicatorMBS
n.tooltip = "Hello"
MsgBox n.tooltip
```

Notes: (Read and Write property)

4.33.102 userInteractionEnabled as Boolean

Plugin Version: 21.4, Platform: macOS, Targets: iOS only.

Function: A Boolean value that determines whether user events are ignored and removed from the event queue.

Notes: When set to false, touch, press, keyboard, and focus events intended for the view are ignored and removed from the event queue. When set to YES, events are delivered to the view normally. The default value of this property is true.

During an animation, user interactions are temporarily disabled for all views involved in the animation, regardless of the value in this property. You can disable this behavior by specifying the `UIViewAnimationOptionAllowUserInteraction` option when configuring the animation.

(Read and Write property)

4.33.103 visibleRect as NSRectMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the portion of the receiver not clipped by its superviews.

Notes: Visibility for this method is defined quite simply and doesn't account for whether other NSView objects (or windows) overlap the receiver or whether the receiver has a window at all. This method returns an empty rectangle if the receiver is effectively hidden.

During a printing operation the visible rectangle is further clipped to the page being imaged.

(Read only property)

4.33.104 `wantsDefaultClipping` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns whether the Application Kit's default clipping provided to `drawRect` implementations is in effect.

Notes: Subclasses may override this method to return false if they want to suppress the default clipping. They may want to do this in situations where drawing performance is critical to avoid the cost of setting up, enforcing, and cleaning up the clip path

A view that overrides this method to refuse the default clipping must either set up whatever clipping it requires or constrain its drawing exactly to the list of rectangles returned by `getRectsBeingDrawn`. Failing to do so could result in corruption of other drawing in the view's window.

(Read only property)

4.33.105 `wantsLayer` as Boolean

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver and its subviews use a Core Animation layer as a backing store.

Notes: True if the view and its subviews should use a Core Animation layer as its backing store, otherwise false.

Requires Mac OS X 10.5.

(Read and Write property)

4.33.106 `wantsRestingTouches` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the view wants resting touches.

Notes: A resting touch occurs when a user rests their thumb on a device (for example, the glass trackpad of a MacBook).

By default, these touches are not delivered and are not included in the event's set of touches. Touches may transition in and out of resting at any time. Unless the view wants `wantsRestingTouches`, `began` / `ended` events are simulated as touches transition from resting to active and vice versa.

In general resting touches should be ignored.

Available in Mac OS X v10.6 and later.

(Read and Write property)

4.33.107 window as NSWindowMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the receiver's window object, or nil if it has none.

Notes: (Read only property)

4.33.108 compositingFilter as variant

Plugin Version: 18.4, Platform: macOS, Targets: Desktop only.

Function: The Core Image filter used to composite the view's contents with its background.

Notes: This property contains the compositing filter stored in the compositingFilter property of the view's layer. If the view does not have a layer, setting the value of this property has no effect.

The default value of this property is nil, which causes content to be rendered without any special compositing effects.

We use variant here to reduce plugin dependencies. Please use with variant containing CIFilterMBS.
(Read and Write computed property)

4.33.109 Constants

Constants

Constant	Value	Description
NSBezelBorder	2	One of the constants to specify the type of a view's border. A concave border that makes the view look sunken.
NSFocusRingTypeDefault	0	One of the constants for the <code>focusringtype</code> property. The default focus ring type for <code>NSView</code> or <code>NSCell</code> .
NSFocusRingTypeExterior	2	One of the constants for the <code>focusringtype</code> property. The standard Aqua focus ring.
NSFocusRingTypeNone	1	One of the constants for the <code>focusringtype</code> property. No focus ring. If you set the focus ring type to this value, <code>NSView</code> will not draw any focus ring.
NSGrooveBorder	3	One of the constants to specify the type of a view's border. A thin border that looks etched around the image.
NSLineBorder	1	One of the constants to specify the type of a view's border. A black line border around the view.
NSNoBorder	0	One of the constants to specify the type of a view's border. No border.
NSViewHeightSizable	16	One of the constants for the <code>autoresizingMask</code> property. The receiver's height is flexible.
NSViewLayerContentsRedrawBeforeViewResize	3	One of the view layer content redraw policy constants.
NSViewLayerContentsRedrawDuringViewResize	2	One of the view layer content redraw policy constants.
NSViewLayerContentsRedrawNever	0	One of the view layer content redraw policy constants.
NSViewLayerContentsRedrawOnSetNeedsDisplay	1	One of the view layer content redraw policy constants.
NSViewMaxXMargin	4	One of the constants for the <code>autoresizingMask</code> property. The right margin between the receiver and its superview is flexible.
NSViewMaxYMargin	32	One of the constants for the <code>autoresizingMask</code> property. The top margin between the receiver and its superview is flexible.
NSViewMinXMargin	1	One of the constants for the <code>autoresizingMask</code> property. The left margin between the receiver and its superview is flexible.
NSViewMinYMargin	8	One of the constants for the <code>autoresizingMask</code> property. The bottom margin between the receiver and its superview is flexible.
NSViewNotSizable	0	One of the constants for the <code>autoresizingMask</code> property. The receiver cannot be resized.
NSViewWidthSizable	2	One of the constants for the <code>autoresizingMask</code> property. The receiver's width is flexible.

Window Order Constants

Constant	Value	Description
NSWindowAbove	1	Moves the window above the indicated window.
NSWindowBelow	-1	Moves the window below the indicated window.
NSWindowOut	0	Moves the window off the screen.

4.34 class `NSViewTooltipMBS`

4.34.1 class `NSViewTooltipMBS`

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: The class for a dynamical `NSView` tooltip.

Notes: Can be used for customized helptags with event to return current text.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.0](#)
- [MBS Xojo Plugins, version 18.0pr7](#)

Xojo Developer Magazine

- [16.2, page 9: News](#)

4.34.2 Methods

4.34.3 Constructor

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: The constructor.

4.34.4 Properties

4.34.5 Text as String

Plugin Version: 18.0, Platform: macOS, Targets: Desktop only.

Function: The text for the tooltip.

Notes: If event is not implemented, we use this default text.

(Read and Write property)

4.34.6 Events

4.34.7 `stringForToolTip(point as NSPointMBS) as string`

Plugin Version: 18.0, Platform: macOS, Targets: .

Function: Queries string for tooltip at position.

Notes: The point allows you to customize the tooltip for the location.
Please return the text to display.

4.35 class NSWindowControllerMBS

4.35.1 class NSWindowControllerMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: An NSWindowController object manages a window, usually a window stored in a nib file.

Notes: Please read Apple's documentation for details:

https://developer.apple.com/library/mac/documentation/Cocoa/Reference/ApplicationKit/Classes/NSWindowController_Class/Reference/Reference.html

Subclass of the NSResponderMBS class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 13.5pr6](#)

4.35.2 Methods

4.35.3 close

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Closes the window if it was loaded.

Notes: Because this method closes the window without asking the user for confirmation, you usually do not invoke it when the Close menu command is chosen. Instead invoke NSWindow's performClose on the receiver's window.

4.35.4 Constructor(win as NSWindowMBS)

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Returns a window controller initialized with a given window.

Notes: window: The window object to manage; can be nil.

Returns a newly initialized window controller.

This method is the designated initializer for NSWindowController.

This initializer is useful when a window has been loaded but no window controller is assigned. The default initialization turns on cascading, sets the shouldCloseDocument flag to NO, and sets the window frame autosave name to an empty string. As a side effect, the created window controller is added as an observer of the NSWindowWillCloseNotifications posted by that window object (which is handled by a private method). If you make the window controller a delegate of the window, you can implement NSWindow's windowShouldClose delegate method.

See also:

- 4.35.5 Constructor(windowNibName as string)

4.35.5 Constructor(windowNibName as string)

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Returns a window controller initialized with a nib file.

Notes: windowNibName: The name of the nib file (minus the ".nib" extension) that archives the receiver's window; cannot be empty.

Sets the owner of the nib file to the receiver. The default initialization turns on cascading, sets the shouldCloseDocument flag to false, and sets the autosave name for the window's frame to an empty string.

See also:

- 4.35.4 Constructor(win as NSWindowMBS)

465

4.35.6 showWindow

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Displays the window associated with the receiver.

Notes: If the window is an NSPanel object and has its becomesKeyOnlyIfNeeded flag set to true, the window is displayed in front of all other windows but is not made key; otherwise it is displayed in front and is made key. This method is useful for menu actions.

4.35.7 synchronizeWindowTitleWithDocumentName

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Synchronizes the displayed window title and the represented filename with the information in the associated document.

Notes: Does nothing if the window controller has no associated document or loaded window. This method queries the window controller's document to get the document's display name and full filename path, then calls windowTitleForDocumentDisplayName to get the display name to show in the window title.

4.35.8 Properties

4.35.9 className as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The name of this NSWindowController class.

Notes: (Read only property)

4.35.10 classPath as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The path of this NSWindowController class.

Notes: Useful for debugging to know what super classes the window controller has.
(Read only property)

4.35.11 shouldCascadeWindows as boolean

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver should necessarily close the associated document when the window it manages is closed.

Notes: True if the receiver necessarily closes the associated document when the window it manages is closed, false otherwise.

If false, the document is closed only when the last remaining window of the document is closed.

The default is false.

(Read and Write property)

4.35.12 shouldCloseDocument as boolean

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Whether the receiver should necessarily close the associated document when the window it manages is closed.

Notes: True if the receiver necessarily closes the associated document when the window it manages is closed, false otherwise.

If false, the document is closed only when the last remaining window of the document is closed.

The default is false.

(Read and Write property)

4.35.13 window as NSWindowMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The window owned by the controller.

Notes: If the window has not yet been loaded, this method attempts to load the window's nib file using loadWindow. Before it loads the window, it invokes windowWillLoad, and if the window controller has a document, it invokes the document's corresponding method windowControllerWillLoadNib (if implemented). After loading the window, this method invokes windowDidLoad and, if there is a document, the NSDocument method windowControllerDidLoadNib (if implemented).

(Read and Write property)

4.35.14 `windowFrameAutosaveName` as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The name under which the frame rectangle of the window owned by the receiver is stored in the defaults database.

Notes: By default, name is an empty string, causing no information to be stored in the defaults database.
(Read and Write property)

4.35.15 `windowNibName` as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Returns the name of the nib file that stores the window associated with the receiver.

Notes: (Read only property)

4.35.16 `windowNibPath` as string

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Returns the full path of the nib file that stores the window associated with the receiver.

Notes: The full path of the nib file that stores the window associated with the receiver; "" if it cannot be located.

(Read only property)

4.36 class NSWindowMBS

4.36.1 class NSWindowMBS

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The Cocoa class for a window.

Example:

```
// add gray layer over Xojo window. Or hide it on second call, show on third call, etc...
```

```
If view = Nil Then
```

```
// view is property in window/module
```

```
Dim win As NSWindowMBS = Self.NSWindowMBS
```

```
view = New NSViewMBS(0, 0, Width, Height)
```

```
win.contentView.AddSubview(view)
```

```
view.wantsLayer = True
```

```
view.autoresizesSubviews = True
```

```
view.autoresizingMask = view.NSViewWidthSizable + view.NSViewHeightSizable
```

```
Dim layer As CALayerMBS = view.layer
```

```
layer.backgroundColor = CGColorMBS.CreateDeviceGray(0.3, 0.5)
```

```
Else
```

```
// show or hide
```

```
view.isHidden = Not view.isHidden
```

```
End If
```

Notes: Subclass of the NSResponderMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 23.3](#)
- [MBS Xojo / Real Studio plug-ins in version 16.5](#)
- [New for Mac OS X 10.10 in MBS Xojo Plugins](#)
- [Please don't call constructors outside new or constructor](#)
- [Lion features for Real Studio](#)
- [Lion arrived](#)

- [Tip of the day: a floating Cocoa HUD window](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Plugins 10.3 Release Notes](#)
- [MonkeyBread Software Releases the MBS Plugins 8.2](#)

Xojo Developer Magazine

- [15.1, page 9: News](#)

4.36.2 Methods

4.36.3 addChildWindow(win as DesktopWindow, order as integer)

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

Function: Adds a given window as a child window of the window.

Notes: win: The child window to order.

order: Either NSWindowAbove: childWindow is ordered immediately in front of the window, or NSWindowBelow: childWindow is ordered immediately behind the window.

After the childWindow is added as a child of the window, it is maintained in relative position indicated by order mode for subsequent ordering operations involving either window. While this attachment is active, moving childWindow will not cause the window to move (as in sliding a drawer in or out), but moving the window will cause childWindow to move.

Note that you should not create cycles between parent and child windows. For example, you should not add window B as child of window A, then add window A as a child of window B.

See also:

- [4.36.4 addChildWindow\(win as NSWindowMBS, order as Integer\)](#) 470
- [4.36.5 addChildWindow\(win as window, order as Integer\)](#) 471

4.36.4 addChildWindow(win as NSWindowMBS, order as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Adds a given window as a child window of the window.

Notes: win: The child window to order.

order: Either NSWindowAbove: childWindow is ordered immediately in front of the window, or NSWindowBelow: childWindow is ordered immediately behind the window.

After the `childWindow` is added as a child of the window, it is maintained in relative position indicated by `order` mode for subsequent ordering operations involving either window. While this attachment is active, moving `childWindow` will not cause the window to move (as in sliding a drawer in or out), but moving the window will cause `childWindow` to move.

Note that you should not create cycles between parent and child windows. For example, you should not add window B as child of window A, then add window A as a child of window B.

See also:

- 4.36.3 `addChildWindow(win as DesktopWindow, order as integer)` 470
- 4.36.5 `addChildWindow(win as window, order as Integer)` 471

4.36.5 `addChildWindow(win as window, order as Integer)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Adds a given window as a child window of the window.

Notes: `win`: The child window to order.

`order`: Either `NSWindowAbove`: `childWindow` is ordered immediately in front of the window, or `NSWindowBelow`: `childWindow` is ordered immediately behind the window.

After the `childWindow` is added as a child of the window, it is maintained in relative position indicated by `order` mode for subsequent ordering operations involving either window. While this attachment is active, moving `childWindow` will not cause the window to move (as in sliding a drawer in or out), but moving the window will cause `childWindow` to move.

Note that you should not create cycles between parent and child windows. For example, you should not add window B as child of window A, then add window A as a child of window B.

See also:

- 4.36.3 `addChildWindow(win as DesktopWindow, order as integer)` 470
- 4.36.4 `addChildWindow(win as NSWindowMBS, order as Integer)` 470

4.36.6 `addTabbedWindow(win as NSWindowMBS, ordered as Integer)`

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Allows creating a group of tabbed windows, or adding a new window to an existing tabbed window group.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
```

```
n.addTabbedWindow(Window2, n.NSWindowAbove)
```

Notes: The 'window' will be added to the receiver's tabbed window group, or create a group if needed. The `tabbingIdentifier` for the entire group should be the same for all the windows, otherwise an exception will be thrown. Use the ordered parameter with "NSWindowAbove" and "NSWindowBelow" to place the new window before or after the receiver's tab. Passing "NSWindowOut" will throw an exception. Currently this method is not animatable, but that may change in the future.

Raises an exception when used on OS X 10.11 and older.

4.36.7 animator as NSWindowMBS

Plugin Version: 10.0, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the proxy object for this window which animates.

Example:

```
dim v as NSWindowMBS // your NSWindowMBS object
```

```
v.alphaValue = 0.5 // switch alpha directly
```

```
v.animator.alphaValue = 0.5 // switch alpha animated
```

4.36.8 areCursorRectsEnabled as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the window's cursor rectangles are enabled.

Notes: Available in Mac OS X v10.0 and later.

4.36.9 attachedSheet as NSWindowMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the sheet attached to the window.

Notes: The sheet attached to the window; nil when the window doesn't have a sheet attached.

Available in Mac OS X v10.1 and later.

4.36.10 autorecalculatesContentBorderThicknessForEdge(edge as Integer) as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window calculates the thickness of a given border automatically.

Notes: edge: Border whose thickness autorecalculation status to set:

NSMaxYEdge: Top border.

NSMinYEdge: Bottom border.

Returns true when the window auto-recalculates the given border's thickness; otherwise, false.

Requires Mac OS X 10.5.

4.36.11 becomeKeyWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Invoked automatically to inform the window that it has become the key window; never invoke this method directly.

Notes: This method reestablishes the window's first responder, sends the becomeKeyWindow message to that object if it responds, and posts an NSWindowDidBecomeKeyNotification to the default notification center.

4.36.12 becomeMainWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Invoked automatically to inform the window that it has become the main window; never invoke this method directly.

4.36.13 cacheImageInRect(r as NSRectMBS)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Stores the window's raster image from a given rectangle expressed in the window's base coordinate system.

Notes: This method allows the window to perform temporary drawing, such as a band around the selection as the user drags the mouse, and to quickly restore the previous image by invoking restoreCachedImage and flushWindowIfNeeded. The next time the window displays, it discards its cached image rectangles. You can also explicitly use discardCachedImage to free the memory occupied by cached image rectangles. aRect is

made integral before caching the image to avoid antialiasing artifacts.

Only the last cached rectangle is remembered and can be restored.

4.36.14 Center

Plugin Version: 8.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the window's location to the center of the screen.

Notes: The window is placed exactly in the center horizontally and somewhat above center vertically. Such a placement carries a certain visual immediacy and importance. This method doesn't put the receiver on-screen, however; use `makeKeyAndOrderFront (show)` to do that.

You typically use this method to place a window—most likely an alert dialog—where the user can't miss it. This method is invoked automatically when a panel is placed on the screen by the `runModalForWindow` method of the `NSApplication` class.

4.36.15 childWindows as NSWindowMBS()

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an array of the window's attached child windows.

4.36.16 ClearFocus

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Clears the focus.

Notes: Moves the focus back to the window.

4.36.17 Close

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Closes the window.

4.36.18 Constructor(w as DesktopWindow)

Plugin Version: 22.1, Platform: macOS, Targets: Desktop only.

Function: Creates a NSWindow for the given Xojo window.

Example:

```
dim w as new NSWindowMBS(window1)
```

```
MsgBox w.Title
```

See also:

- 4.36.19 Constructor(w as window) 475
- 4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false) 475

4.36.19 Constructor(w as window)

Plugin Version: 9.4, Platform: macOS, Targets: Desktop only.

Function: Creates a NSWindow for the given Xojo window.

Example:

```
dim w as new NSWindowMBS(window1)
```

```
MsgBox w.Title
```

Notes: In plugin version 9.4 and newer this works only with Cocoa windows.

In plugin version 9.7 and newer you can use it also for Carbon windows.

See also:

- 4.36.18 Constructor(w as DesktopWindow) 475
- 4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false) 475

4.36.20 Constructor(x as Double, y as Double, w as Double, h as Double, styleMask as Integer, BackingStoreType as Integer = 0, deferCreation as boolean = false, canBecomeKeyWindow as boolean = false)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The constructor to create a new Cocoa Windows.

Notes: *x,y,w,h*:

Location and size of the window's content area in screen coordinates. Note that the window server limits window position coordinates to $\mp 16,000$ and sizes to 10,000.

styleMask:

The window's style. Either it can be `NSBorderlessWindowMask`, or it can contain any of the options described in the constants, combined using the `bitwiseOR` function. Borderless windows display none of the usual peripheral elements and are generally useful only for display or caching purposes; you should normally not need to create them. Also, note that a window's style mask should include `NSTitledWindowMask` if it includes any of the others.

bufferingType:

Specifies how the drawing done in the window is buffered by the window device, and possible values are described in "Constants."

deferCreation:

Specifies whether the window server creates a window device for the window immediately. When true, the window server defers creating the window device until the window is moved onscreen. All display messages sent to the window or its views are postponed until the window is created, just before it's moved onscreen.

Initialized `NSWindow` object.

This method is the designated initializer for the `NSWindow` class.

Deferring the creation of the window improves launch time and minimizes the virtual memory load on the window server.

The new window creates a view to be its default content view. You can replace it with your own object by using the `ContentView` property.

Parameter `canBecomeKeyWindow` (in 11.3 plugin) controls whether we use a special `NSWindow` subclass which returns true for the `canBecomeKeyWindow` function. This way you can create windows which can take key focus.

See also:

- 4.36.18 `Constructor(w as DesktopWindow)` 475
- 4.36.19 `Constructor(w as window)` 475

4.36.21 `contentBorderThicknessForEdge(edge as Integer) as Double`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates the thickness of a given border of the window.

Notes: edge: The border whose thickness to get:

NSMaxYEdge: Top border.

NSMinYEdge: Bottom border.

Requires Mac OS X 10.5.

4.36.22 contentRectForFrameRect(windowFrame as NSRectMBS) as NSRectMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the window's content rectangle with a given frame rectangle.

Example:

```
dim w as NSWindowMBS = window1.NSWindowMBS
dim fr as NSRectMBS = w.frame
dim cr as NSRectMBS = w.contentRectForFrameRect(fr)
```

```
MsgBox "Title height: "+str(fr.Height-cr.Height)
```

Notes: windowFrame: The frame rectangle for the window expressed in screen coordinates.

Returns the window's content rectangle, expressed in screen coordinates, with windowFrame.

The window uses its current style mask in computing the content rectangle. See Window Style Mask constants for a list of style mask values. The main advantage of this instance-method counterpart to contentRectForFrameRect (With styleMask) is that it allows you to take toolbars into account when converting between content and frame rectangles. (The toolbar is not included in the content rectangle.)

See also:

- 4.36.23 contentRectForFrameRect(windowFrame as NSRectMBS, styleMask as UInt32) as NSRectMBS
477

4.36.23 contentRectForFrameRect(windowFrame as NSRectMBS, styleMask as UInt32) as NSRectMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the content rectangle used by a window with a given frame rectangle and window style.

Notes: windowFrame: The frame rectangle for the window expressed in screen coordinates.

`windowStyle`: The window style for the window. See constants for a list of style mask values.

Returns the content rectangle, expressed in screen coordinates, used by the window with `windowFrame` and `windowStyle`.

When a `NSWindowMBS` instance is available, you should use `contentRectForFrameRect` instead of this method.

See also:

- 4.36.22 `contentRectForFrameRect(windowFrame as NSRectMBS) as NSRectMBS` 477

4.36.24 `convertBaseToScreen(p as NSPointMBS) as NSPointMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a given point from the window's base coordinate system to the screen coordinate system.

4.36.25 `convertScreenToBase(p as NSPointMBS) as NSPointMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Converts a given point from the screen coordinate system to the window's base coordinate system.

4.36.26 `dataWithEPSInsideRect(r as NSRectMBS) as Memoryblock`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns EPS data that draws the region of the window within a given rectangle.

Notes: This data can be placed on a pasteboard, written to a file, or used to create an `NSImage` object.

4.36.27 `dataWithPDFInsideRect(r as NSRectMBS) as Memoryblock`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns PDF data that draws the region of the window within a given rectangle.

Notes: This data can be placed on a pasteboard, written to a file, or used to create an `NSImage` object.

4.36.28 `deminiaturize`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: This action method deminimizes the receiver.

Notes: Invoke this method to programmatically deminimize a minimized window in the Dock.

4.36.29 disableCursorRects

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Disables all cursor rectangle management within the window.

Notes: Use this method when you need to do some special cursor manipulation and you don't want the Application Kit interfering.

4.36.30 disableFlushWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Disables the flushWindow method for the window.

Notes: If the window is buffered, disabling flushWindow prevents drawing from being automatically flushed by the NSView display... methods from the window's backing store to the screen. This method permits several views to be drawn before the results are shown to the user.

Flushing should be disabled only temporarily, while the window's display is being updated. Each disableFlushWindow message must be paired with a subsequent enableFlushWindow message. Invocations of these methods can be nested; flushing isn't reenabled until the last (unnested) enableFlushWindow message is sent.

4.36.31 disableScreenUpdatesUntilFlush

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Disables the window's screen updates until the window is flushed.

Notes: This method can be invoked to synchronize hardware surface flushes with the window's flushes. The window immediately disables screen updates using the NSDisableScreenUpdates function and reenables screen updates when the window flushes. Sending this message multiple times during a window update cycle has no effect.

Available in Mac OS X v10.4 and later.

4.36.32 disableSnapshotRestoration

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Disable snapshot restoration.

Notes: While snapshot restoration is disabled, the window will not be snapshotted for restorable state. Available in OS X v10.7 and later.

4.36.33 `discardCachedImage`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Discards all of the window's cached image rectangles.

Notes: An `NSWindow` object automatically discards its cached image rectangles when it displays.

4.36.34 `discardCursorRects`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Invalidates all cursor rectangles in the window.

Notes: This method is invoked by `resetCursorRects` to clear out existing cursor rectangles before resetting them. You shouldn't invoke it in the code you write, but you might want to override it to change its behavior.

4.36.35 `display`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Passes a `display` message down the window's view hierarchy, thus redrawing all views within the window, including the frame view that draws the border, title bar, and other peripheral elements.

Notes: You rarely need to invoke this method. `NSWindow` objects normally record which of their views need display and display them automatically on each pass through the event loop.

4.36.36 `displayIfNeeded`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Passes a `displayIfNeeded` message down the window's view hierarchy, thus redrawing all views that need to be displayed, including the frame view that draws the border, title bar, and other peripheral elements.

Notes: This method is useful when you want to modify some number of views and then display only the ones that were modified.

You rarely need to invoke this method. `NSWindow` objects normally record which of their views need display and display them automatically on each pass through the event loop.

4.36.37 dockTile as Variant

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Provides the application's Dock tile.

Notes: Available in Mac OS X v10.5 and later.

Returns NSDockTileMBS object for the dock tile.

4.36.38 dragImage(image as NSImageMBS, viewLocation as NSPointMBS, offset as NSSizeMBS, NSEvent as NSEventMBS, pboard as NSPasteboardMBS, source as NSViewMBS, slideFlag as boolean)

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Begins a dragging session.

Notes: image: The object to be dragged.

imageLocation: Location of the image's bottom-left corner in the window's coordinate system. It determines the placement of the dragged image under the pointer.

offset: The pointer's location relative to the mouse-down location. Not used in OS X v10.4 and later.

NSEvent: The left-mouse down event that triggered the dragging operation.

pasteboard: The pasteboard that holds the data to be transferred to the destination.

source: The object serving as the controller of the dragging operation. It must conform to the NSDraggingSource protocol.

slideBack: Specifies whether the drag image should slide back to imageLocation if it's rejected by the drag destination. Pass true to specify slide back behavior or false to specify that it should not.

This method should be invoked only from within a view's implementation of the mouseDown or mouseDragged methods (which overrides the version defined in NSResponder class). Essentially the same as the NSView method of the same name, except that imageLocation is given in the NSWindow object's base coordinate system.

4.36.39 enableCursorRects

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Reenables cursor rectangle management within the window after a disableCursorRects message.

4.36.40 enableFlushWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Reenables the `flushWindow` method for the window after it was disabled through a previous `disableFlushWindow` message.

4.36.41 `enableSnapshotRestoration`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Enable snapshot restoration.

Notes: While snapshot restoration is enabled, the window will be snapshotted for restorable state. Available in OS X v10.7 and later.

4.36.42 `endEditingFor(anObject as object = nil)`

Plugin Version: 13.4, Platform: macOS, Targets: Desktop & iOS.

Function: Forces the field editor to give up its first responder status and prepares it for its next assignment.

Notes: `anObject`: The object that is using the window's field editor.

If the field editor is the first responder, it's made to resign that status even if its `resignFirstResponder` method returns false. This registration forces the field editor to send a `textDidEndEditing` message to its delegate. The field editor is then removed from the view hierarchy, its delegate is set to `nil`, and it's emptied of any text it may contain.

This method is typically invoked by the object using the field editor when it's finished. Other objects normally change the first responder by simply using `makeFirstResponder`, which allows a field editor or other object to retain its first responder status if, for example, the user has entered an invalid value. The `endEditingFor:` method should be used only as a last resort if the field editor refuses to resign first responder status. Even in this case, you should always allow the field editor a chance to validate its text and take whatever other action it needs first. You can do this by first trying to make the `NSWindow` object the first responder.

4.36.43 `fieldEditor(createFlag as boolean = True, forObject as object = nil) as Variant`

Plugin Version: 13.4, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the window's field editor, creating it if requested.

Notes: Returns `NSTextMBS` object. Returned as `Variant` to reduce plugin dependencies.

`createWhenNeeded`: If true, creates a field editor if one doesn't exist; if false, does not create a field editor. A freshly created `NSWindow` object doesn't have a field editor. After a field editor has been created for a window, the `createWhenNeeded` argument is ignored. By passing false for `createWhenNeeded` and testing

the return value, however, you can predicate an action on the existence of the field editor.

forObject: A text-displaying object for which the delegate (in `windowWillReturnFieldEditor`) assigns a custom field editor. Pass `nil` to get the default field editor, which can be the `NSWindow` field editor or a custom field editor returned by the delegate.

Returns the field editor for the designated object (`anObject`) or, if `anObject` is `nil`, the default field editor. Returns `nil` if `createFlag` is `false` and if the field editor doesn't exist.

The field editor is a single `NSTextView` object that is shared among all the controls in a window for light text-editing needs. It is automatically instantiated when needed, and it can be used however your application sees fit. Typically, the field editor is used by simple text-bearing objects—for example, an `NSTextField` object uses its window's field editor to display and manipulate text. The field editor can be shared by any number of objects, and so its state may be constantly changing. Therefore, it shouldn't be used to display text that demands sophisticated layout (for this you should create a dedicated `NSTextView` object).

The field editor may be in use by some view object, so be sure to properly dissociate it from that object before actually using it yourself (the appropriate way to do this is illustrated in the description of `endEditingFor:`). Once you retrieve the field editor, you can insert it in the view hierarchy, set a delegate to interpret text events, and have it perform whatever editing is needed. Then, when it sends a `textDidEndEditing` message to the delegate, you can get its text to display or store and remove the field editor using `endEditingFor`.

The window's delegate can substitute a custom field editor in place of the window's field editor by implementing `windowWillReturnFieldEditor`. The custom field editor can become the default editor (common to all text-displaying objects) or specific to a particular text-displaying object (`anObject`). The window sends this message to its delegate with itself and `anObject` as the arguments; if the delegate returns a non-`nil` value, the window returns that object instead of its field editor in `fieldEditor`. However, note the following:

If the window's delegate is identical to `anObject`, `windowWillReturnFieldEditor` isn't sent to the delegate. The object returned by the delegate method, though it may become first responder, does not become the window's default field editor. Other objects continue to use the window's default field editor.

4.36.44 firstResponder as NSResponderMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window's first responder.

Notes: The first responder is usually the first object in a responder chain to receive an event or action message. In most cases, the first responder is a view object in that the user selects or activates with the mouse or keyboard.

You can use the `firstResponder` method in custom subclasses of responder classes (`NSWindow`, `NSApplication`, `NSView`, and subclasses) to determine if an instance of the subclass is currently the first responder. You can also use it to help locate a text field that currently has first-responder status. For more on this

subject, see Event Handling Basics (on developer.apple.com).

4.36.45 flushWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Flushes the window's offscreen buffer to the screen if the window is buffered and flushing is enabled.

Notes: Does nothing for other display devices, such as a printer. This method is automatically invoked by the `NSWindow` `display` and `displayIfNeeded` methods and the corresponding `NSView` `display` and `displayIfNeeded` methods.

4.36.46 flushWindowIfNeeded

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Flushes the window's offscreen buffer to the screen if flushing is enabled and if the last flush-Window message had no effect because flushing was disabled.

Notes: To avoid unnecessary flushing, use this method rather than `flushWindow` to flush an `NSWindow` object after flushing has been reenabled.

4.36.47 frameRectForContentRect(windowContent as NSRectMBS) as NSRectMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the window's frame rectangle with a given content rectangle.

Notes: `windowContent`: The content rectangle for the window expressed in screen coordinates.

Returns the window's frame rectangle, expressed in screen coordinates, with `windowContent`.

The window uses its current style mask in computing the frame rectangle. See "Window Style Masks" for a list of style mask values. The major advantage of this instance-method counterpart to `frameRectForContentRect` (with `styleMask`) is that it allows you to take toolbars into account when converting between content and frame rectangles. (The toolbar is included in the frame rectangle but not the content rectangle.)

See also:

- 4.36.48 `frameRectForContentRect(windowContentRect as NSRectMBS, styleMask as UInt32) as NSRectMBS`

4.36.48 `frameRectForContentRect(windowContentRect as NSRectMBS, styleMask as UInt32) as NSRectMBS`

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the frame rectangle used by a window with a given content rectangle and window style.

Notes: `windowContentRect`: The content rectangle for a window expressed in screen coordinates.

`windowStyle`: The window style for the window. See Window Style Mask constants for a list of style mask values.

Returns the frame rectangle, expressed in screen coordinates, used by the window with `windowContentRect` and `windowStyle`.

When a `NSWindowMBS` instance is available, you should use `frameRectForContentRect` instead of this method.

See also:

- 4.36.47 `frameRectForContentRect(windowContent as NSRectMBS) as NSRectMBS` 484

4.36.49 `GetFrame(byref left as Double, byref top as Double, byref width as Double, byref height as Double)`

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Asks for the current window dimensions.

Notes: The point 0/0 is on the left bottom position.

4.36.50 `gState as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the window's graphics state object.

Notes: This graphics state is used by default for all `NSView` objects in the window's view hierarchy, but individual views can be made to use their own with the `NSView` method `allocateGState`.

4.36.51 `Hide`

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Hides the window.

4.36.52 `inLiveResize` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window is being resized by the user.

Notes: Available in Mac OS X v10.6 and later.

4.36.53 `invalidateCursorRectsForView(View as NSViewMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Marks as invalid the cursor rectangles of a given `NSView` object in the window's view hierarchy, so they'll be set up again when the window becomes key (or immediately if the window is key).

4.36.54 `invalidateRestorableState`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Method that you may call to indicate that the restorable state is invalid.

Notes: At some point in the future, `encodeRestorableStateWithCoder:` will be called to encode the restorable state. You should not override this method.

4.36.55 `invalidateShadow`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Invalidates the window shadow so that it is recomputed based on the current window shape.

4.36.56 `keyDown(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Handles a given keyboard event that may need to be interpreted as changing the key view or triggering a keyboard equivalent.

Notes: event: The keyboard event to process.

4.36.57 `makeFirstResponder(r as NSResponderMBS)` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Attempts to make a given responder the first responder for the window.

Example:

```
dim MySearchField as NSViewMBS // your control to move focus to
dim w as new NSWindowMBS(window1)
call w.makeFirstResponder(MySearchField)
```

Notes: responder: The responder to set as the window's first responder. nil makes the window its first responder.

Returns true when the operation is successful; otherwise, false.

If responder isn't already the first responder, this method first sends a resignFirstResponder message to the object that is the first responder. If that object refuses to resign, it remains the first responder, and this method immediately returns false. If the current first responder resigns, this method sends a becomeFirstResponder message to responder. If responder does not accept first responder status, the NSWindow object becomes first responder; in this case, the method returns true even if responder refuses first responder status.

If responder is nil, this method still sends resignFirstResponder to the current first responder. If the current first responder refuses to resign, it remains the first responder and this method immediately returns false. If the current first responder returns true from resignFirstResponder, the window is made its own first responder and this method returns true.

The Application Kit framework uses this method to alter the first responder in response to mouse-down events; you can also use it to explicitly set the first responder from within your program. The responder object is typically an NSView object in the window's view hierarchy. If this method is called explicitly, first send acceptsFirstResponder to responder, and do not call makeFirstResponder if acceptsFirstResponder returns false.

Use setInitialFirstResponder to set the first responder to be used when the window is brought onscreen for the first time.

4.36.58 makeKeyAndOrderFront

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the window to the front of the screen list, within its level, and makes it the key window; that is, it shows the window.

Notes: More or less the same as Xojo's show command on the window class.

4.36.59 makeKeyWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Makes the window the key window.

4.36.60 makeMainWindow

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Makes the window the main window.

4.36.61 mergeAllWindows

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Merges all windows into one window.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.mergeAllWindows
```

Notes: Raises an exception when used on OS X 10.11 and older.

4.36.62 minFrameWidthWithTitle(WindowTitle as string, styleMask as UInt32) as Double

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the minimum width a window's frame rectangle must have for it to display a title, with a given window style.

Example:

```
// shows 84.77588
MsgBox str(NSWindowMBS.minFrameWidthWithTitle("Hello World",0))
```

Notes: windowTitle: The title for the window.

windowStyle: The window style for the window. See Window Style Mask constants for a list of style mask values.

The minimum width of the window's frame, using `windowStyle`, in order to display `windowTitle`.

4.36.63 miniaturize

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: This action method removes the receiver from the screen list and displays the minimized window in the Dock.

4.36.64 moveTabToNewWindow

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Moves current tab to a new window.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.moveTabToNewWindow
```

Notes: Raises an exception when used on OS X 10.11 and older.

4.36.65 NSDockWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for the doc. (Deprecated. There is no replacement.)

4.36.66 NSFloatingWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: Useful for floating palettes.

4.36.67 NSMainMenuWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: Reserved for the application's main menu.

4.36.68 NSModalPanelWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for a modal panel.

4.36.69 NSNormalWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The default level for NSWindow objects.

4.36.70 NSPopUpMenuWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for a pop-up menu.

4.36.71 NSScreenSaverWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for a screen saver.

4.36.72 NSStatusWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for a status window.

4.36.73 NSSubmenuWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: Reserved for submenus. Synonymous with NSTornOffMenuWindowLevel, which is preferred.

4.36.74 NSTornOffMenuWindowLevel as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the standard window levels in Mac OS X.

Notes: The level for a torn-off menu. Synonymous with NSSubmenuWindowLevel.

4.36.75 NSWindowDidBecomeKeyNotification as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted whenever an NSWindow object becomes the key window.

The notification object is the NSWindow object that has become key. This notification does not contain a userInfo dictionary.

4.36.76 NSWindowDidBecomeMainNotification as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted whenever an NSWindow object becomes the main window.

The notification object is the NSWindow object that has become main. This notification does not contain a userInfo dictionary.

4.36.77 NSWindowDidChangeScreenNotification as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted whenever a portion of an NSWindow object's frame moves onto or off of a screen.

The notification object is the `NSWindow` object that has changed screens. This notification does not contain a `userInfo` dictionary.

This notification is not sent in Mac OS X versions earlier than 10.4.

4.36.78 `NSWindowDidChangeScreenProfileNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever the display profile for the screen containing the window changes.

This notification is sent only if the window returns true from `displaysWhenScreenProfileChanges`. This notification may be sent when a majority of the window is moved to a different screen (whose profile is also different from the previous screen) or when the `ColorSync` profile for the current screen changes.

The notification object is the `NSWindow` object whose profile changed. This notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.4 and later.

4.36.79 `NSWindowDidDeminiaturizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object is deminimized.

The notification object is the `NSWindow` object that has been deminimized. This notification does not contain a `userInfo` dictionary.

4.36.80 `NSWindowDidEndLiveResizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted after the user resizes a window.

This notification is sent only once for a series of window resize operations.

The notification object is the NSWindow object that was resized. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.6 and later.

4.36.81 NSWindowDidEndSheetNotification as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted whenever an NSWindow object closes an attached sheet.

The notification object is the NSWindow object that contained the sheet. This notification does not contain a userInfo dictionary.

4.36.82 NSWindowDidEnterFullScreenNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window entered full screen mode.

The notification object is the NSWindow object entered full screen mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.83 NSWindowDidEnterVersionBrowserNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window will exit full screen mode.

The notification object is the NSWindow object that will exit full screen mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.84 `NSWindowDidEnterFullScreenNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted when the window will exit full screen mode.

The notification object is the `NSWindow` object that will exit full screen mode. This notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.7 and later.

4.36.85 `NSWindowDidEnterVersionBrowserNotification` as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted when the window did exit version browser mode.

The notification object is the `NSWindow` object that did exit version browser mode. This notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.7 and later.

4.36.86 `NSWindowDidExposeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever a portion of a nonretained `NSWindow` object is exposed, whether by being ordered in front of other windows or by other windows being removed from in front of it.

The notification object is the `NSWindow` object that has been exposed. The `userInfo` dictionary contains the following information:

Key	Value
<code>NSExposedRect</code>	The rectangle that has been exposed (<code>NSRect</code>).

4.36.87 `NSWindowDidMiniaturizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object is minimized.

The notification object is the `NSWindow` object that has been minimized. This notification does not contain a `userInfo` dictionary.

4.36.88 `NSWindowDidMoveNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object is moved.

The notification object is the `NSWindow` object that has moved. This notification does not contain a `userInfo` dictionary.

Note: This notification is sent when the window that moved didn't also change size. See `NSWindowDidResizeNotification` for more information.

4.36.89 `NSWindowDidResignKeyNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object resigns its status as key window.

The notification object is the `NSWindow` object that has resigned its key window status. This notification does not contain a `userInfo` dictionary.

4.36.90 `NSWindowDidResignMainNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object resigns its status as main window.

The notification object is the `NSWindow` object that has resigned its main window status. This notification does not contain a `userInfo` dictionary.

4.36.91 `NSNotificationDidResizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSNotification` object's size changes.

The notification object is the `NSNotification` object whose size has changed. This notification does not contain a `userInfo` dictionary.

4.36.92 `NSNotificationDidUpdateNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSNotification` object receives an update message.

The notification object is the `NSNotification` object that received the update message. This notification does not contain a `userInfo` dictionary.

4.36.93 `NSNotificationWillBeginSheetNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSNotification` object is about to open a sheet.

The notification object is the `NSNotification` object that is about to open the sheet. This notification does not contain a `userInfo` dictionary.

4.36.94 `NSNotificationWillCloseNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSNotification` object is about to close.

The notification object is the `NSNotification` object that is about to close. This notification does not contain a `userInfo` dictionary.

4.36.95 NSWindowDidEnterFullScreenNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window will enter full screen mode.

The notification object is the NSWindow object will enter full screen mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.96 NSWindowDidEnterVersionBrowserNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window will enter version browser mode.

The notification object is the NSWindow object that will enter version browser mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.97 NSWindowWillExitFullScreenNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window will exit full screen mode.

The notification object is the NSWindow object that will exit full screen mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.98 NSWindowWillExitVersionBrowserNotification as string

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the NSNotification* classes.

Notes: Posted when the window will exit version browser mode.

The notification object is the NSWindow object that will exit version browser mode. This notification does not contain a userInfo dictionary.

Available in Mac OS X v10.7 and later.

4.36.99 `NSWindowWillMiniaturizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object is about to be minimized.

The notification object is the `NSWindow` object that is about to be minimized. This notification does not contain a `userInfo` dictionary.

4.36.100 `NSWindowWillMoveNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted whenever an `NSWindow` object is about to move.

The notification object is the `NSWindow` object that is about to move. This notification does not contain a `userInfo` dictionary.

4.36.101 `NSWindowWillStartLiveResizeNotification` as string

Plugin Version: 10.3, Platform: macOS, Targets: Desktop & iOS.

Function: One of the notification strings you can use with the `NSNotification*` classes.

Notes: Posted before the user resizes a window.

This notification is sent only once for a series of window resize operations.

The notification object is the `NSWindow` object that is about to be live resized. This notification does not contain a `userInfo` dictionary.

Available in Mac OS X v10.6 and later.

4.36.102 `orderBack`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the window to the back of its level in the screen list, without changing either the key window or the main window.

4.36.103 orderFront

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the window to the front of its level in the screen list, without changing either the key window or the main window.

4.36.104 orderFrontRegardless

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Moves the window to the front of its level, even if its application isn't active, without changing either the key window or the main window.

Notes: Normally an `NSWindow` object can't be moved in front of the key window unless it and the key window are in the same application. You should rarely need to invoke this method; it's designed to be used when applications are cooperating in such a way that an active application (with the key window) is using another application to display data.

Available in Mac OS X v10.0 and later.

4.36.105 orderOut

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the window from the screen list, which hides the window.

Notes: More or less the same as Xojo's hide command on the window class.

If the window is the key or main window, the `NSWindow` object immediately behind it is made key or main in its place. Calling the `orderOut` method causes the window to be removed from the screen, but does not cause it to be released. See the `close` method for information on when a window is released.

The default animation based on the window type will be used when the window is ordered out unless it has been modified by the `setAnimationBehavior` method.

4.36.106 PerformClose

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: This action method simulates the user clicking the close button by momentarily highlighting the button and then closing the window.

Notes: Same as if the user clicks the close button.

So if the button is disabled or closing is not permitted by the window, it will do nothing.

4.36.107 performMiniaturize

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: This action method simulates the user clicking the minimize button by momentarily highlighting the button, then minimizing the window.

Notes: If the receiver doesn't have a minimize button or can't be minimized for some reason, the system emits the alert sound.

4.36.108 performWindowDragWithEvent(event as NSEventMBS)

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Starts a drag based on the specified mouse-down event.

Notes: event: The original mouse-down event received by the application or a view. If you don't have a NSEvent object, please pass the one from `NSApplicationMBS.currentEvent` here.

Your application (or a view) can call this method after receiving and examining a mouse-down event. Upon examination of the event, a view may allow that portion of the window to start a window drag and can hand off the work to the Window Server process by calling this method. Doing so allows the window to participate in space switching and other system features.

This method returns right away, and a mouse-up event may not get sent.

4.36.109 performZoom

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: This action method simulates the user clicking the zoom box by momentarily highlighting the button and then zooming the window.

Notes: If the receiver doesn't have a zoom box or can't be zoomed for some reason, the computer beeps.

4.36.110 print

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: This action method runs the Print panel, and if the user chooses an option other than canceling, prints the window (its frame view and all subviews).

4.36.111 registerForDraggedTypes(Types() as string)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Registers a give set of pasteboard types as the pasteboard types the window will accept as the destination of an image-dragging session.

Notes: Types: An array of the pasteboard types the window will accept as the destination of an image-dragging session.

Registering an NSWindow object for dragged types automatically makes it a candidate destination object for a dragging session. NSWindow has a default implementation for many of the methods in the NSDraggingDestination informal protocol. The default implementation forwards each message to the delegate if the delegate responds to the selector of the message. The messages forwarded this way are draggingEntered:, draggingUpdated:, draggingExited:, prepareForDragOperation:, performDragOperation:, and concludeDragOperation:.

Xojo with Cocoa target implements the methods listed above. So this method is useful to change the allowed types for the case the framework has a bug.

4.36.112 removeChildWindow(win as NSWindowMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Detaches a given child window from the window.

Notes: win: The child window to detach.

See also:

- 4.36.113 removeChildWindow(win as window) 501

4.36.113 removeChildWindow(win as window)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Detaches a given child window from the window.

Notes: win: The child window to detach.

See also:

- 4.36.112 removeChildWindow(win as NSWindowMBS) 501

4.36.114 removeFrameUsingName(name as string)

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Removes the frame data stored under a given name from the application’s user defaults.

4.36.115 `resetCursorRects`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Clears the window’s cursor rectangles and the cursor rectangles of the `NSView` objects in its view hierarchy.

Notes: Invokes `discardCursorRects` to clear the window’s cursor rectangles, then sends `resetCursorRects` to every `NSView` object in the window’s view hierarchy.

This method is typically invoked by the `NSApplication` object when it detects that the key window’s cursor rectangles are invalid. In program code, it’s more efficient to invoke `invalidateCursorRectsForView`.

4.36.116 `resignKeyWindow`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Invoked automatically when the window resigns key window status; never invoke this method directly.

Notes: This method sends `resignKeyWindow` to the window’s first responder, sends `windowDidResignKey` to the window’s delegate, and posts an `NSWindowDidResignKeyNotification` to the default notification center.

4.36.117 `resignMainWindow`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Invoked automatically when the window resigns main window status; never invoke this method directly.

Notes: This method sends `windowDidResignMain` to the window’s delegate and posts an `NSWindowDidResignMainNotification` to the default notification center.

4.36.118 `resizeFlags as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the flags field of the event record for the mouse-down event that initiated the resizing session.

Notes: A mask indicating which of the modifier keys was held down when the mouse-down event occurred.

The flags are listed in NSEvent object's modifierFlags method description.

This method is valid only while the window is being resized

You can use this method to constrain the direction or amount of resizing. Because of its limited validity, this method should only be invoked from within an implementation of the delegate method windowWillResize.

4.36.119 restoreCachedImage

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Splices the window's cached image rectangles, if any, back into its raster image (and buffer if it has one), undoing the effect of any drawing performed within those areas since they were established using cacheImageInRect.

Notes: You must invoke flushWindow after this method to guarantee proper redisplay. An NSWindow object automatically discards its cached image rectangles when it displays.

4.36.120 runToolbarCustomizationPalette

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: The action method for the "Customize Toolbar..." menu item.

4.36.121 saveFrameUsingName(s as String)

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Saves the window's frame rectangle in the user defaults system under a given name.

Notes: With the companion method setFrameUsingName, you can save and reset an NSWindow object's frame over various launches of an application. The default is owned by the application and stored under the name "NSWindow Frame frameName". See NSUserDefaults for more information.

4.36.122 selectKeyViewFollowingView(view as NSViewMBS)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Makes key the view that follows the given view.

Notes: Sends the nextValidKeyView message to referenceView and, if that message returns an NSView object, invokes makeFirstResponder with the returned object.

4.36.123 `selectKeyViewPrecedingView(view as NSViewMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Makes key the view that follows the given view.

Notes: Sends the `nextValidKeyView` message to `referenceView` and, if that message returns an `NSView` object, invokes `makeFirstResponder` with the returned object.

4.36.124 `selectNextKeyView`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: This action method searches for a candidate next key view and, if it finds one, invokes `makeFirstResponder` to establish it as the first responder.

Notes: The candidate is one of the following (searched for in this order):

- The current first responder's next valid key view, as returned by the `nextValidKeyView` method of `NSView`
- The object designated as the window's initial first responder (using `setInitialFirstResponder`) if it returns true to an `acceptsFirstResponder` message
- Otherwise, the initial first responder's next valid key view, which may end up being nil

4.36.125 `selectNextTab`

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Selects next tab.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.selectNextTab
```

Notes: Raises an exception when used on OS X 10.11 and older.

4.36.126 `selectPreviousKeyView`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: This action method searches for a candidate previous key view and, if it finds one, invokes `makeFirstResponder` to establish it as the first responder.

Notes: The candidate is one of the following (searched for in this order):

- The current first responder's previous valid key view, as returned by the `previousValidKeyView` method of `NSView`
- The object designated as the window's initial first responder (using `setInitialFirstResponder`) if it returns true to an `acceptsFirstResponder` message
- Otherwise, the initial first responder's previous valid key view, which may end up being nil

4.36.127 `selectPreviousTab`

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Selects previous tab.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.selectPreviousTab
```

Notes: Raises an exception when used on OS X 10.11 and older.

4.36.128 `sendEvent(e as NSEventMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: This action method dispatches mouse and keyboard events sent to the window by the `NSApplication` object.

Notes: Never invoke this method directly. A right mouse-down event in a window of an inactive application is not delivered to the corresponding `NSWindow` object. It is instead delivered to the `NSApplication` object through a `sendEvent:` message with a window number of 0.

4.36.129 `setAutorecalculatesContentBorderThickness(flag as boolean, edge as Integer)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether the window calculates the thickness of a given border automatically.

Notes: `flag`: If true, the window calculates the thickness of the edge automatically; if false, it does not.

`edge`: The border whose thickness auto-recalculation status to set:

NSMaxYEdge: Top border.
 NSMinYEdge: Bottom border.

Special Considerations

Turning off a border's auto-recalculation status sets its border thickness to 0.0.

In a non-textured window calling `setAutorecalculatesContentBorderThickness` passing `NSMaxYEdge` will raise an exception. It is only valid to set the content border thickness of the top edge in a textured window.

Requires Mac OS X 10.5.

4.36.130 `setBottomCornerRounded(flag as boolean)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the bottom corner to be round.

Notes: This is an undocumented function in the `NSWindow` class.

4.36.131 `setContentBorderThickness(thickness as Double, edge as Integer)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the thickness of a given border of the window.

Example:

```
dim w as NSWindowMBS = window1.NSWindowMBS
```

```
w.setContentBorderThickness(30, w.NSMinYEdge) // bottom 30 pixel
w.setContentBorderThickness(20, w.NSMaxYEdge) // top 30 pixel. Window must be metal for this
```

Notes: thickness: The thickness for edge, in points.

edge: The border whose thickness to set:

NSMaxYEdge: Top border.
 NSMinYEdge: Bottom border.

In a non-textured window calling `setContentBorderThickness` passing `NSMaxYEdge` will raise an exception. It is only valid to set the content border thickness of the top edge in a textured window.

The `contentBorder` does not include the titlebar or toolbar, so a textured window that just wants the gra-

dient in the titlebar and toolbar should have a `contentBorderThickness` of 0 for `NSMaxYEdge`.

Requires Mac OS X 10.5.

4.36.132 `setContentSize(size as NSSizeMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the size of the window's content view to a given size, which is expressed in the window's base coordinate system.

Notes: This size in turn alters the size of the `NSWindow` object itself. Note that the window server limits window sizes to 10,000; if necessary, be sure to limit `aSize` relative to the frame rectangle.

4.36.133 `setFrame(frameRect as NSRectMBS)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the origin and size of the window's frame rectangle according to a given frame rectangle, thereby setting its position and size onscreen.

Notes: `frameRect`: The frame rectangle for the window, including the title bar.

Note that the window server limits window position coordinates to $\neg\pm 16,000$ and sizes to 10,000.
See also:

- 4.36.134 `setFrame(frameRect as NSRectMBS, display as boolean)` 507
- 4.36.135 `setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)` 508
- 4.36.136 `SetFrame(left as Double, top as Double, width as Double, height as Double)` 508

4.36.134 `setFrame(frameRect as NSRectMBS, display as boolean)`

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the origin and size of the window's frame rectangle according to a given frame rectangle, thereby setting its position and size onscreen.

Notes: The point 0/0 is on the bottom left position of the main screen.

`display`: Specifies whether the window redraws the views that need to be displayed. When true the window sends a `displayIfNeeded` message down its view hierarchy, thus redrawing all views.

See also:

- 4.36.133 `setFrame(frameRect as NSRectMBS)` 507
- 4.36.135 `setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)` 508

- 4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double) 508

4.36.135 setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the origin and size of the window's frame rectangle, with optional animation, according to a given frame rectangle, thereby setting its position and size onscreen.

Notes: The point 0/0 is on the bottom left position of the main screen.

If animated is true, the change is animated.

See also:

- 4.36.133 setFrame(frameRect as NSRectMBS) 507
- 4.36.134 setFrame(frameRect as NSRectMBS, display as boolean) 507
- 4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double) 508

4.36.136 setFrame(left as Double, top as Double, width as Double, height as Double)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the origin and size of the window's frame rectangle according to a given frame rectangle, thereby setting its position and size onscreen.

Notes: The point 0/0 is on the bottom left position of the main screen.

See also:

- 4.36.133 setFrame(frameRect as NSRectMBS) 507
- 4.36.134 setFrame(frameRect as NSRectMBS, display as boolean) 507
- 4.36.135 setFrame(frameRect as NSRectMBS, display as boolean, animated as boolean) 508

4.36.137 setFrameAutosaveName(name as String) as boolean

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the name used to automatically save the window's frame rectangle in the defaults system to a given name.

Notes: Returns true when the frame name is set successfully; false when frameName is being used as an autosave name by another NSWindow object in the application (in which case the window's old name remains in effect).

If `frameName` isn't the empty string (""), the window's frame is saved as a user default (as described in `saveFrameUsingName`) each time the frame changes.

When the window has an autosave name, its frame data is written whenever the frame rectangle changes.

If there is a frame rectangle previously stored for `frameName` in the user defaults, the window's frame is set to this frame rectangle. That is, when you call this method with a previously used `frameName`, the window picks up the previously saved setting. For example, if you call `setFrameAutosaveName` for a window that is already onscreen, this method could cause the window to move to a different screen location. For this reason, it is generally better to call this method before the window is visible on screen.

Keep in mind that a window controller may change the window's position when it displays it if window cascading is turned on. To preclude the window controller from changing a window's position from the one saved in the defaults system, you must send `setShouldCascadeWindows(false)` to the window controller.

4.36.138 `setFrameFromString(s as String)`

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the window's frame rectangle from a given string representation.

Notes: If the window is not resizable, this method will not resize the window. The frame is constrained according to the window's minimum and maximum size settings. This method can cause a `windowWillResize` event.

4.36.139 `setFrameOrigin(point as NSPointMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Positions the bottom-left corner of the window's frame rectangle at a given point in screen coordinates.

Notes: Note that the window server limits window position coordinates to $\neg\pm 16,000$.

4.36.140 `setFrameTopLeftPoint(point as NSPointMBS)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Positions the top-left corner of the window's frame rectangle at a given point in screen coordinates.

Notes: Note that the window server limits window position coordinates to $\neg\pm 16,000$; if necessary, adjust `aPoint` relative to the window's lower-left corner to account for this limit.

4.36.141 `setFrameUsingName(name as String, force as boolean = false) as boolean`

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Sets the window's frame rectangle by reading the rectangle data stored under a given name from the defaults system.

Notes: Returns true when `frameName` is read and the frame is set successfully; otherwise, false.

The frame is constrained according to the window's minimum and maximum size settings. This method causes a `windowWillResize` event.

4.36.142 `setRestorationClass`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the class to use to acquire a matching window object during subsequent launches.

Notes: The plugin provides a class to restore windows. See `NSWindowRestoreHandlerMBS` class.

The restoration class of a window is responsible for recreating not just the window but any other objects needed to manage the window. Therefore, the restoration class must be able to create (or find existing instances of) all of these objects at launch time in your application.

If you mark your windows as restorable, you must associate a restoration class with them.
Available in OS X v10.7 and later.

4.36.143 `setTitleWithRepresentedFile(filename as folderitem)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets a given path as the window's title, formatting it as a file-system path, and records this path as the window's associated filename using `setRepresentedFilename`.

4.36.144 `setTitleWithRepresentedFilename(filename as string)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Sets a given path as the window's title, formatting it as a file-system path, and records this path as the window's associated filename using `setRepresentedFilename`.

4.36.145 Show

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Shows the window.

4.36.146 standardWindowButton(button as Integer) as Variant

Plugin Version: 12.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the window button of a given window button kind in the window's view hierarchy.

Example:

```
// another way to disable close button on Cocoa
dim n as NSButtonMBS = window1.NSWindowMBS.standardWindowButton(NSWindowMBS.NSWindow-
CloseButton)
n.isEnabled = false

// or move to to the right
dim r as NSRectMBS = n.frame
n.frame = NSMakeRectMBS(r.x+100, r.y, r.Width, r.Height)
```

Notes: button: The kind of standard window button to return.

Returns Window button in the window's view hierarchy of the kind identified by windowButtonKind; nil when such button is not in the window's view hierarchy.

Declared as Variant for reduced plugin dependencies.

Button constants: NSWindowCloseButton, NSWindowDocumentIconButton, NSWindowMiniaturizeButton, NSWindowToolbarButton and NSWindowZoomButton.

See also:

- 4.36.147 standardWindowButton(button as Integer, StyleMask as Integer) as Variant 511

4.36.147 standardWindowButton(button as Integer, StyleMask as Integer) as Variant

Plugin Version: 12.1, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a new instance of a given standard window button, sized appropriately for a given window style.

Notes: button: The kind of standard window button to return.

StyleMask: The window style for which windowButtonKind is to be sized. See "Window Style Masks" for

the list of allowable values.

Returns the new window button of the kind identified by `windowButtonKind`; nil when no such button kind exists.

The caller is responsible for adding the button to the view hierarchy and for setting the target to be the window.

Declared as Variant for reduced plugin dependencies.

Button constants: `NSWindowCloseButton`, `NSWindowDocumentIconButton`, `NSWindowMiniaturizeButton`, `NSWindowToolbarButton` and `NSWindowZoomButton`

See also:

- 4.36.146 `standardWindowButton(button as Integer) as Variant` 511

4.36.148 `stringWithSavedFrame as String`

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: Returns a string representation of the window's frame rectangle.

Notes: A string representation of the window's frame rectangle in a format that can be used with a later `setFrameFromString` method.

4.36.149 `tabbedWindows as NSWindowMBS()`

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the entire group (stack) of windows that are all visually shown together in one virtual tabbed window and associated with this particular window.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
dim windows() as NSWindowMBS = n.tabbedWindows
break // check in debugger
```

Notes: Operations can then be done on each window, as necessary. For instance, iterating over each window in the group and calling `performClose:` will close the entire stack. The result will be nil when the window is not tabbed at all (not showing a tab bar), and non-nil with at least one object when the tab bar is shown. The order of items in the array is the same order as the tabs visually shown (leading to trailing).

Raises an exception when used on OS X 10.11 and older.

4.36.150 toggleFullScreen

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Toggles fullscreen mode for this window.

Example:

```
dim w as new NSWindowMBS(window1)
```

```
// set window to have fullscreen
```

```
w.collectionBehavior = BitwiseOr(w.collectionBehavior, NSWindowMBS.NSWindowCollectionBehaviorFullScreen-  
Primary)
```

```
// and switch to fullscreen
```

```
w.toggleFullScreen
```

Notes: Available on Mac OS X 10.7 or later.

If an application supports fullscreen, it should add a menu item to the View menu with toggleFullScreen as the action.

This method does not much if you don't mark a window to be the primary fullscreen window.

4.36.151 toggleTabBar

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Toggles tab bar.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS  
n.toggleTabBar
```

Notes: Raises an exception when used on OS X 10.11 and older.

4.36.152 toggleToolBarShown

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The action method for the "Hide Toolbar" menu item (which alternates with "Show Toolbar").

4.36.153 toolbarview as NSViewMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop & iOS.

Function: Convenience function to find the toolbar view on the window.

Example:

```
dim v as NSViewMBS = window1.NSWindowMBS.toolbarview
dim m as NSMenuMBS = v.Menu
dim i as NSMenuItemMBS = m.Item(0)
MsgBox i.title
```

Notes: Returns nil if there is no toolbar view.

4.36.154 unregisterDraggedTypes

Plugin Version: 11.1, Platform: macOS, Targets: Desktop & iOS.

Function: Unregisters the window as a possible destination for dragging operations.

4.36.155 update

Plugin Version: 8.3, Platform: macOS, Targets: Desktop & iOS.

Function: Updates the window.

Notes: The NSWindow implementation of this method does nothing more than post an NSWindowDidUpdateNotification notification to the default notification center. A subclass can override this method to perform specialized operations, but it should send an update message to super just before returning. For example, the NSMenu class implements this method to disable and enable menu commands.

An NSWindow object is automatically sent an update message on every pass through the event loop and before it's displayed onscreen. You can manually cause an update message to be sent to all visible NSWindow objects through the NSApplication updateWindows method.

4.36.156 useOptimizedDrawing(value as boolean)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether the window is to optimize focusing and drawing when displaying its views.

Notes: value: If true, the window will optimize focusing and drawing for its views; if false, it will not, in which case, the window does not preserve the Z-ordering of overlapping views when an object explicitly

sends `lockFocus` to a view and draws directly to it, instead of using the AppKit standard display mechanism.

The optimizations may prevent sibling subviews from being displayed in the correct order—which matters only if the subviews overlap. You should always set `optimizedDrawing` to true when there are no overlapping subviews within the window. The default is false.

4.36.157 WindowHandle as Integer

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The Window handle.

Notes: Can be used with the `CarbonWindowsEventsMBS` class.
(for events like open and close)

4.36.158 windowNumberAtPoint(x as Double, y as Double, belowWindowWithWindowNumber as Integer = 0) as Integer

Plugin Version: 13.0, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the number of the frontmost window that would be hit by a `MouseDown` at the screen location point.

Notes: `belowWindowWithWindowNumber` can be specified to exclude a given window along with all windows above it, and may belong to any application. If no windows are to be excluded, specify 0 for `belowWindowWithWindowNumber`. The `windowNumber` returned may correspond to a window in another application.

Requires Mac OS X 10.6 or newer.

4.36.159 windowNumbersWithOptions(options as Integer = 0) as Integer()

Plugin Version: 13.0, Platform: macOS, Targets: Desktop & iOS.

Function: Returns an array of integers containing `windowNumbers` for all visible windows satisfying options.

Example:

```
dim windowNumbers1() as Integer
dim windowNumbers2() as Integer
dim windowNumbers3() as Integer
```

```
// To get an array of windowNumbers visible on the current space and belonging to the calling application:
```

```
windowNumbers1 = NSWindowMBS.windowNumbersWithOptions(0)
```

```
// To get an array of windowNumbers visible on any space and belonging to any application:
```

```
windowNumbers2 = NSWindowMBS.windowNumbersWithOptions(NSWindowMBS.NSWindowNumberListAllApplications+NSWindowMBS.NSWindowNumberListAllSpaces)
// To get an array of windowNumbers visible on any space and belonging to the calling application:
windowNumbers3 = NSWindowMBS.windowNumbersWithOptions(NSWindowMBS.NSWindowNumberListAllSpaces)
```

Notes: In no options are specified, only visible windows belonging to the calling application and on the active space are included. If options include `NSWindowNumberListAllApplications`, visible windows belonging to all applications are included. If options include `NSWindowNumberListAllSpaces`, visible windows on all spaces are included.

4.36.160 zoom

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: This action method toggles the size and location of the window between its standard state (provided by the application as the "best" size to display the window's data) and its user state (a new size and location the user may have set by moving or resizing the window).

4.36.161 Properties

4.36.162 acceptsMouseMovedEvents as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver is to accept mouse-moved events.

Notes: True to have the receiver accept mouse-moved events (and to distribute them to its responders); false to not accept such events.

(Read and Write property)

4.36.163 allowsAutomaticWindowTabbing as Boolean

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Allows automatic window tabbing when the value is true.

Example:

```
// disable automatic tabbing
NSWindowMBS.allowsAutomaticWindowTabbing = false
```

Notes: By default, this will be set to true, but applications can explicitly opt out of all automatic tabbing

by setting it to false, and can still adopt explicit window tabbing, if desired.
 Raises an exception when used on OS X 10.11 and older.
 (Read and Write property)

4.36.164 allowsConcurrentViewDrawing as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether threading of view drawing is enabled for this window.

Notes: Whether threading of view drawing should be enabled for this window. Defaults to true. When this is set to true, AppKit's view system is allowed to perform drawRect activity for the window's views on threads other than the main thread, for views that have canDrawConcurrently = true. When this is set to false, the window's views will be drawn serially as on 10.5 and earlier, even though some of the views may have canDrawConcurrently = true.

Available on Mac OS X 10.6.
 (Read and Write property)

4.36.165 allowsToolTipsWhenApplicationIsInactive as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether this window displays tooltips even when the application is in the background.

Notes: Default is false. Set to true to allow a window to display tooltips even when the application is in the background. Note that, enabling tooltips in an inactive application will cause the app to do work any time the mouse passes over the window. This can degrade system performance.

(Read and Write property)

4.36.166 alphaValue as Double

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The receiver's alpha value.

Example:

```
if TargetCocoa then
dim w as new NSWindowMBS(window1)
w.alphaValue=0.5
else
MsgBox "this sample requires Xojo Cocoa Target"
end if
```

Notes: (Read and Write property)

4.36.167 `animationBehavior` as Integer

Plugin Version: 11.2, Platform: macOS, Targets: Desktop & iOS.

Function: Provides for per-window control over automatic `orderFront/orderOut` animation behaviors added in 10.7.

Example:

```
dim w as new NSWindowMBS(window1)
```

```
w.animationBehavior = NSWindowMBS.NSWindowAnimationBehaviorDocumentWindow
```

```
Title = str(W.animationBehavior)
```

Notes: Can be set to `NSWindowAnimationBehaviorNone` to disable AppKit's automatic animations for a given window, or to one of the other non-Default `NSWindowAnimationBehavior` values to override AppKit's automatic inference of appropriate animation behavior based on the window's apparent type.

Available on Mac OS X 10.7 or later.

(Read and Write property)

4.36.168 `aspectRatio` as `NSSizeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window's aspect ratio, which constrains the size of its frame rectangle to integral multiples of this ratio when the user resizes it.

Notes: (Read and Write property)

4.36.169 `Autodisplay` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the window automatically displays views that need to be displayed.

Notes: Automatic display typically occurs on each pass through the event loop.

Available in Mac OS X v10.0 and later.

(Read and Write property)

4.36.170 backgroundColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window background color.

Notes: (Read and Write property)

4.36.171 backingLocation as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates the window's backing store location.

Notes: The location of the window's backing store. See "Constants" for possible values. Available in Mac OS X v10.5 and later.

(Read only property)

4.36.172 backingScaleFactor as Double

Plugin Version: 12.3, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the backing scale factor.

Notes: Returns 2.0 for high resolution scaled display modes, and 1.0 for all other cases.

There are some scenarios where an application that is resolution-aware may want to reason on its own about the display environment it is running in.

It is important to note that this number returned by this method does not represent anything concrete, such as pixel density or physical size, since it can vary based on the configured display mode. For example, the display may be in a mirrored configuration that is still high resolution scaled, resulting in pixel geometry that may not match the native resolution of the display device.

Note: For almost all common cases, developers should avoid using the `backingScaleFactor` as an input to layout or drawing calculations. Developers should instead use the backing coordinate space conversion methods instead, as the resulting code will more likely work consistently and correctly under both low and high resolution operation.

For apps which are not enabled for retina support, the function returns 1. So you only see 2 here if app is Cocoa, display is retina and info.plist has the `NSHighResolutionCapable` key.

(Read only property)

4.36.173 backingType as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window’s backing store type.

Notes: Use constants like this:

NSBackingStoreRetained	= 0	The window uses a buffer, but draws directly to the screen where possible and to the buffer for obscured portions. You should not use this mode. It combines the limitations of NSBackingStoreNonretained with the memory use of NSBackingStoreBuffered. The original NeXTSTEP implementation was an interesting compromise that worked well with fast memory mapped framebuffers on the CPU bus—something that hasn’t been in general use since around 1994. These tend to have performance problems. In Mac OS X 10.5 and later, requests for retained windows will result in the window system creating a buffered window, as that better matches actual use. Available in Mac OS X v10.0 and later.
NSBackingStoreNonretained	= 1	The window draws directly to the screen without using any buffer. You should not use this mode. It exists primarily for use in the original Classic Blue Box. It does not support Quartz drawing, alpha blending, or opacity. Moreover, it does not support hardware acceleration, and interferes with system-wide display acceleration. If you use this mode, your application must manage visibility region clipping itself, and manage repainting on visibility changes. Available in Mac OS X v10.0 and later.
NSBackingStoreBuffered	= 2	The window renders all drawing into a display buffer and then flushes it to the screen. You should use this mode. It supports hardware acceleration, Quartz drawing, and takes advantage of the GPU when possible. It also supports alpha channel drawing, opacity controls, using the compositor. Available in Mac OS X v10.0 and later.

(Read and Write property)

4.36.174 canBecomeKeyWindow as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window can become the key window.

Notes: Attempts to make the window the key window are abandoned if this method returns false. The NSWindow implementation returns true if the window has a title bar or a resize bar, or false otherwise.

(Read only property)

4.36.175 canBecomeMainWindow as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window can become the application's main window.

Notes: Attempts to make the window the main window are abandoned if this method returns false. The NSWindow implementation returns true if the window is visible, is not an NSPanel object, and has a title bar or a resize mechanism. Otherwise it returns false.

(Read only property)

4.36.176 canBecomeVisibleWithoutLogin as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window can be displayed at the login window.

Notes: Default: false.

Available in Mac OS X v10.5 and later.

(Read and Write property)

4.36.177 canHide as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether the window can be hidden when its application becomes hidden (during execution of the NSApplication hide method).

Notes: (Read and Write property)

4.36.178 canStoreColor as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window has a depth limit that allows it to store color values.

Notes: (Read only property)

4.36.179 className as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: The name of this NSWindow class.

Notes: (Read only property)

4.36.180 `classPath` as string

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: The path of this `NSView` class.

Notes: Useful for debugging to know what super classes the window has.
(Read only property)

4.36.181 `collectionBehavior` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Identifies the window's behavior in window collections.

Notes: Available in Mac OS X v10.5 and later.
(Read and Write property)

4.36.182 `colorSpace` as `NSColorSpaceMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The window's color space.

Notes: Available in Mac OS X v10.6 and later.
(Read and Write property)

4.36.183 `contentAspectRatio` as `NSSizeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The aspect ratio (height in relation to width) of the window's content view, constraining the dimensions of its content rectangle to integral multiples of that ratio when the user resizes it.

Notes: Available in Mac OS X v10.3 and later
(Read and Write property)

4.36.184 `contentLayoutRect` as `NSRectMBS`

Plugin Version: 21.4, Platform: macOS, Targets: Desktop & iOS.

Function: The area inside the window that is for non-obscured content, in window coordinates.

Notes: Typically, the area represented by this property is the same as the frame of the `contentView`. However, for windows with `NSFullSizeContentViewWindowMask` set, there needs to be a way to determine the portion that is not under the toolbar. The `contentLayoutRect` property contains the portion of the layout that is not obscured under the toolbar.

(Read only property)

4.36.185 contentMaxSize as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum size of the window's content view.

Notes: Available in Mac OS X v10.3 and later.

(Read and Write property)

4.36.186 contentMinSize as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum size of the window's content view.

Notes: Available in Mac OS X v10.3 and later.

(Read and Write property)

4.36.187 contentResizeIncrements as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window's content-view resizing increments.

Notes: Available in Mac OS X v10.3 and later.

(Read and Write property)

4.36.188 contentView as NSViewMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The receiver's content view, the highest accessible NSView object in the receiver's view hierarchy.

Notes: (Read and Write property)

4.36.189 currentEvent as NSEventMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the event currently being processed by the application, by invoking NSApplication's currentEvent method.

Notes: Returns the event being processed by the application.

(Read only property)

4.36.190 `deepestScreen` as `NSScreenMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the deepest screen the window is on (it may be split over several screens).

Notes: The deepest screen the window is on; nil when the window is offscreen.

(Read only property)

4.36.191 `depthLimit` as `Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the depth limit of the window.

Notes: The value returned can be examined with the Application Kit functions `NSPlanarFromDepth`, `NSColorSpaceFromDepth`, `NSBitsPerSampleFromDepth`, and `NSBitsPerPixelFromDepth`.

(Read and Write property)

4.36.192 `displaysWhenScreenProfileChanges` as `boolean`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window context should be updated when the screen profile changes or when the window moves to a different screen.

Notes: Returns true when the window context should be updated when the screen profile changes or when the window moves to a different screen; otherwise, false.

The default value is false.

Available in Mac OS X v10.4 and later.

(Read and Write property)

4.36.193 `frame` as `NSRectMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window's frame rectangle.

Notes: (Read only property)

4.36.194 hasDynamicDepthLimit as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window's depth limit can change to match the depth of the screen it's on.

Notes: True when the window has a dynamic depth limit; otherwise, false.

(Read and Write property)

4.36.195 hasShadow as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver has a shadow.

Notes: True when the receiver has a shadow, false when it doesn't.

(Read and Write property)

4.36.196 Height as Double

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The height of the color panel in pixel.

Notes: (Read and Write property)

4.36.197 hidesOnDeactivate as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver is removed from the screen when its application becomes inactive.

Notes: True when the receiver is removed from the screen when its application is deactivated; false if it remains onscreen.

The default for NSWindow is false; the default for NSPanel is true.

(Read and Write property)

4.36.198 identifier as string

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: A string that identifies this user interface item.

Example:

```
dim w as new NSWindowMBS(window1)
```

```
w.identifier = "window1"  
MsgBox w.identifier
```

Notes: It should be set to a unique value on NSViews when they are intended to be used inside a view-based NSTableView. Identifiers should be unique per-window. For programmatically created user interface items, you would typically set this value in code after creating a control but before adding it to a window. You may also want to set an identifier on a window, after creating it programmatically, to identify the window easily when it is reopened. You should not change the identifier after a control is added to a window. Identifiers beginning with an underscore are reserved for the system. In framework classes that implement this protocol, the accessor methods are not intended to be overridden.

To help avoid collision of identifiers, it is recommended that identifiers use the same prefix as is used for the framework or application. For example, identifiers for standard AppKit interface items, such as the open panel, will begin with "NS".

The slash '/', backslash '\', and colon ':' characters are reserved and should not be used in identifiers. (Read and Write property)

4.36.199 ignoresMouseEvents as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the receiver is transparent to mouse events.

Notes: True when the receiver is transparent to mouse events, otherwise false.

(Read and Write property)

4.36.200 initialFirstResponder as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The view that's made first responder the first time the window is placed onscreen.

Notes: (Read and Write property)

4.36.201 isDocumentEdited as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the receiver's document has been edited.

Notes: True when the receiver's document has been edited; false otherwise. Initially, by default, NSWindow objects are in the "not edited" state. (Read and Write property)

4.36.202 isExcludedFromWindowsMenu as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver's title is omitted from the application's Windows menu.

Notes: True to specify that the receiver is to be omitted from the application's Windows menu; false to specify otherwise. (Read and Write property)

4.36.203 isFlushWindowDisabled as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window's flushing ability is disabled.

Notes: (Read only property)

4.36.204 `isKeyWindow` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window is the key window for the application.

Notes: (Read only property)

4.36.205 `isMainWindow` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window is the application's main window.

Notes: (Read only property)

4.36.206 `isMiniaturized` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver is minimized.

Notes: A minimized window is removed from the screen and replaced by a image, icon, or button that represents it, called the counterpart.

(Read only property)

4.36.207 `isMovableByWindowBackground` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the receiver is movable by clicking and dragging anywhere in its background.

Notes: True when the window is movable by clicking and dragging anywhere in its background, otherwise false.

A window with a style mask of `NSTexturedBackgroundWindowMask` is movable by background by default. Sheets and drawers cannot be movable by window background.

Available in Mac OS X v10.2 and later.

Works in Xojo 2011r3, but not in 2011r4.

(Read and Write property)

4.36.208 isOnActiveSpace as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window is on the currently active space.

Notes: For visible windows, this method indicates whether the window is currently visible on the active space. For offscreen windows, it indicates whether ordering the window onscreen would cause it to be on the active space.

Available in Mac OS X v10.6 and later.

(Read only property)

4.36.209 isOneShot as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window device the window manages is freed when it's removed from the screen list.

Notes: (Read and Write property)

4.36.210 isOpaque as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver is opaque.

Notes: True when the receiver is opaque; false otherwise.

(Read and Write property)

4.36.211 isSheet as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window has ever run as a modal sheet.

Notes: Sheets are created using the NSPanel subclass.

Available in Mac OS X v10.1 and later.

(Read only property)

4.36.212 isZoomed as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: A Boolean value that indicates whether the receiver is in a zoomed state.

Notes: (Read only property)

4.36.213 Left as Double

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The horizontal position of the color panel in pixel.

Notes: (Read and Write property)

4.36.214 Level as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The window level of the receiver.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.Level = n.NSFloatingWindowLevel
```

Notes: (Read and Write property)

4.36.215 maxSize as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The maximum size to which the window's frame (including its title bar) can be sized.

Notes: The maximum size to which the window's frame (including its title bar) can be sized either by the user or by the setFrame... methods other than setFrame.

(Read and Write property)

4.36.216 miniwindowImage as Variant

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The custom miniaturized window image of the receiver.

Notes: The miniaturized window image is the image displayed in the Dock when the window is minimized.

If you did not assign a custom image to the window, this method returns nil.

Value is declares as Variant to minimize plugin dependencies, but should be NSImageMBS.

(Read and Write property)

4.36.217 `miniwindowTitle` as String

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The title displayed in the receiver's minimized window.

Notes: (Read and Write property)

4.36.218 `minSize` as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The minimum size to which the window's frame (including its title bar) can be sized.

Notes: The minimum size to which the window's frame (including its title bar) can be sized either by the user or by the `setFrame...` methods other than `setFrame`.

(Read and Write property)

4.36.219 `Movable` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the window can be moved by clicking in its title bar or background.

Notes: `setMovableByWindowBackground`, called with the argument `true`, is ignored by a window that returns `false` from `isMovable`. If a window returns `false`, that means it can only be dragged between spaces in F8 mode, and its relative screen position is always preserved. Note that a resizable window may still be resized, and the window frame may be changed programmatically. A non-movable window will not be moved or resized by the system in response to a display reconfiguration. Applications may choose to enable application-controlled window dragging after disabling user-initiating dragging by handling the `mouseDown/mouseDragged/mouseUp` sequence in `sendEvent` in an `NSWindow` subclass.

Available in Mac OS X v10.6 and later.

(Read and Write property)

4.36.220 `parentWindow` as NSWindowMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The parent window to which the window is attached as a child.

Notes: Available in Mac OS X v10.2 and later.

(Read and Write property)

4.36.221 `preferredBackingLocation` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The preferred location for the window's backing store.

Notes: Available in Mac OS X v10.5 and later.

(Read and Write property)

4.36.222 `preservesContentDuringLiveResize` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the window tries to optimize user-initiated resize operations by preserving the content of views that have not changed.

Notes: Returns true if the window tries to optimize live resize operations by preserving the content of views that have not moved; otherwise, false.

Discussion

When live-resize optimization is active, the window redraws only those views that moved (or do not support this optimization) during a live resize operation.

See `preservesContentDuringLiveResize` in `NSView` for additional information on how to support this optimization.

Available in Mac OS X v10.4 and later.

(Read and Write property)

4.36.223 `preventsApplicationTerminationWhenModal` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the window prevents application termination when modal.

Notes: Available in Mac OS X v10.6 and later.

(Read and Write property)

4.36.224 `representedFile` as folderitem

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The pathname of the file the window represents.

Notes: (Read and Write property)

4.36.225 representedFilename as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The pathname of the file the window represents.

Notes: (Read and Write property)

4.36.226 representedURL as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The URL of the file the window represents.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim n as new NSWindowMBS(window1)
n.representedURL = f.URLPath
```

```
MsgBox n.representedURL
```

Notes: The URL for the file the window represents.

When the URL specifies a path, the window shows an icon in its title bar, as described in Table 1.

Title bar document icon display:

Filepath	Document icon
Empty	None.
Specifies a nonexistent file	Generic.
Specifies an existent file	Specific for the file's type.

You can customize the file icon in the tile bar with the following code:

```
window.standardWindowButton(NSWindowDocumentIconButton).Image=theImage
```

When the URL identifies an existing file, the window's title offers a pop-up menu showing the path components of the URL. (The user displays this menu by Command-clicking the title.) The behavior and contents of this menu can be controlled with `shouldPopUpDocumentPathMenu`.

Available in Mac OS X v10.5 and later.
(Read and Write property)

4.36.227 `resizeIncrements` as `NSSizeMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The window's resizing increments.
Notes: (Read and Write property)

4.36.228 `Restorable` as `boolean`

Plugin Version: 13.2, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies whether the window configuration is preserved between application launches.
Notes: value: Specify true if you want the window to be preserved or false if you do not want it preserved.

Windows should be preserved between launch cycles to maintain interface continuity for the user. During subsequent launch cycles, the system tries to recreate the window and restore its configuration to the preserved state. Configuration data is updated as needed and saved automatically by the system.

If you enable preservation for a given window, you should also specify a restoration class for the window using the `setRestorationClass` method.
Available in OS X v10.7 and later.
(Read and Write property)

4.36.229 `screen` as `NSScreenMBS`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the screen the window is on.
Notes: The screen where most of the window is on; nil when the window is offscreen.

When the window is partly on one screen and partly on another, the screen where most of it lies is returned.
(Read only property)

4.36.230 `sharingType` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: The level of access other processes have to the window's content.

Notes: Available in Mac OS X v10.5 and later.

(Read and Write property)

4.36.231 `showsResizeIndicator` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver's resize indicator is visible

Notes: True to show it, false to hide it.

This method does not affect whether the receiver is resizable.

(Read and Write property)

4.36.232 `showsToolbarButton` as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the receiver shows the toolbar control button.

Notes: True to display the toolbar control button; false to hide the button.

If the window does not have a toolbar, this method has no effect.

(Read and Write property)

4.36.233 `styleMask` as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: The receiver's style mask, indicating what kinds of control items it displays.

Example:

```
dim p as new IKPictureTakerMBS
p.styleMask = BitwiseAnd(p.styleMask, &hFD) // disable close button
```

Notes: See the information about the style mask in constants below. An `NSWindow` object's style is set when the object is initialized. Once set, it can't be changed.

constants:

<code>NSBorderlessWindowMask = 0</code>	The window displays none of the usual peripheral elements. Useful only for display or caching purposes.
<code>NSTitledWindowMask = 1</code>	The window displays a title bar.
<code>NSClosableWindowMask = 2</code>	The window displays a close button.
<code>NSMiniaturizableWindowMask = 4</code>	The window displays a minimize button.
<code>NSResizableWindowMask = 8</code>	The window displays a resize control.
<code>NSTexturedBackgroundWindowMask = 256</code>	The window displays with a metal-textured background. Additionally, the window may be moved by clicking and dragging anywhere in the window background. A bordered window with this mask gets rounded bottom corners.

(Read and Write property)

4.36.234 SubTitle as String

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Secondary text that may be displayed adjacent to or below the primary title depending on the configuration of the window.

Notes: A value of empty string will remove the subtitle from the window layout.

(Read and Write property)

4.36.235 tabbingIdentifier as String

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Windows with the same `tabbingIdentifier` will have the ability to be tabbed together when a window is being shown.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.tabbingIdentifier = "test"
MsgBox n.tabbingIdentifier
```

Notes: This allows aggregation of similar windows. By default, the `tabbingIdentifier` will be generated based on inherit window properties, such as the window class name, the delegate class name, the window controller class name, and some additional state. Windows can be explicitly made to group together by using the same `tabbingIdentifier`.

Raises an exception when used on OS X 10.11 and older.

(Read and Write property)

4.36.236 tabbingMode as Integer

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Get and set the tabbing mode for this window.

Example:

```
dim n as NSWindowMBS = window1.NSWindowMBS
n.tabbingMode = n.NSWindowTabbingModeDisallowed
MsgBox str(n.tabbingMode) // shows 2
```

Notes: This should be set before a window is shown. The default value is NSWindowTabbingModeAutomatic. When the value is NSWindowTabbingModeAutomatic, the system will look at the userTabbingPreference and automatically tab windows together based on the tabbingIdentifier, when it is appropriate to do so.

Raises an exception when used on OS X 10.11 and older.

(Read and Write property)

4.36.237 Title as String

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The title of the color panel window.

Example:

```
if TargetCocoa then
dim w as new NSWindowMBS(window1)

MsgBox w.title

else
MsgBox "this sample requires Xojo Cocoa Target"
end if
```

Notes: (Read and Write property)

4.36.238 titlebarAppearsTransparent as Boolean

Plugin Version: 15.1, Platform: macOS, Targets: Desktop & iOS.

Function: Whether titlebar appears transparent.

Notes: When True, the titlebar doesn't draw its background, allowing all buttons to show through, and

”click through” to happen. In general, this is only useful when `NSFullSizeContentViewWindowMask` is set.

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

4.36.239 `titlebarSeparatorStyle` as Integer

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies the style of separator displayed between the window’s titlebar and content.

Notes: The default value is `NSTitlebarSeparatorStyleAutomatic`. Changing this value will override any preference made by `NSSplitViewItem`.

(Read and Write property)

4.36.240 `titleVisibility` as Integer

Plugin Version: 14.3, Platform: macOS, Targets: Desktop & iOS.

Function: The title visibility.

Notes: Can be `NSWindowTitleVisible`, `NSWindowTitleHidden` or `NSWindowTitleHiddenWhenActive`.

Available on Mac OS X 10.10 and newer.

(Read and Write property)

4.36.241 `toolbar` as Variant

Plugin Version: 11.3, Platform: macOS, Targets: Desktop & iOS.

Function: The window’s toolbar.

Notes: Value is a `NSToolbarMBS` object.

Returned as Variant to reduce plugin dependencies.

(Read and Write property)

4.36.242 `toolbarStyle` as Integer

Plugin Version: 20.3, Platform: macOS, Targets: Desktop & iOS.

Function: Specifies how the titlebar area of the window should appear when the window displays an `NSToolbar`.

Example:

`Dim w as NSWindowMBS = window1.NSWindowMBS`

```
w.toolbarStyle = NSWindowMBS.NSWindowToolbarStyleUnified
```

Notes: (Read and Write property)

4.36.243 Top as Double

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The vertical position of the color panel in pixel.

Notes: In the Cocoa world this is the distance from the bottom of the screen.
(Read and Write property)

4.36.244 userTabbingPreference as Integer

Plugin Version: 16.5, Platform: macOS, Targets: Desktop & iOS.

Function: Returns the user's tabbing preference as set in System Preferences.

Example:

```
MsgBox str(NSWindowMBS.userTabbingPreference)
```

Notes: This value should be queried anytime a new window is made to see if the user wants to automatically show it in tabs.

Can be `NSWindowUserTabbingPreferenceManual`, `NSWindowUserTabbingPreferenceAlways` or `NSWindowUserTabbingPreferenceInFullScreen`.

Raises an exception when used on OS X 10.11 and older.

(Read only property)

4.36.245 viewsNeedDisplay as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether any of the window's views need to be displayed.

Notes: (Read and Write property)

4.36.246 Visible as boolean

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: Whether the color panel is currently visible.

Notes: Setting to false calls Hide and setting to true calls show.
(Read only property)

4.36.247 Width as Double

Plugin Version: 8.2, Platform: macOS, Targets: Desktop & iOS.

Function: The width of the color panel in pixel.

Notes: (Read and Write property)

4.36.248 windowController as NSWindowControllerMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: The window's window controller.

Notes: (Read and Write property)

4.36.249 windowNumber as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop & iOS.

Function: Provides the window number of the receiver's window device.

Example:

```
if TargetCocoa then
dim w as new NSWindowMBS(window1)

MsgBox str(w.windowNumber)

else
MsgBox "this sample requires Xojo Cocoa Target"
end if
```

Notes: (Read only property)

4.36.250 worksWhenModal as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop & iOS.

Function: Indicates whether the window is able to receive keyboard and mouse events even when some other window is being run modally.

Notes: True if the window is able to receive keyboard and mouse events even when some other window is being run modally; otherwise, false.

The `NSWindow` implementation of this method returns false. Only subclasses of `NSPanel` should override this default.

(Read only property)

4.36.251 `frameAutosaveName` as string

Plugin Version: 15.2, Platform: macOS, Targets: Desktop & iOS.

Function: The name used to automatically save the window's frame rectangle data in the defaults system.

Notes: (Read and Write computed property)

4.36.252 Constants

Constants

Constant	Value	Description
<code>NSBorderlessWindowMask</code>	0	One of the constants you can use to specify the style when creating a window. The window displays none of the usual peripheral elements for display or caching purposes.
<code>NSClosableWindowMask</code>	2	One of the constants you can use to specify the style when creating a window. The window displays a close button.
<code>NSDirectSelection</code>	0	One of the constants specify the direction a window is currently selecting the key view using <code>keyViewSelectionDirection</code> . The window isn't traversing the key view loop.
<code>NSFullScreenWindowMask</code>	16384	One of the constants you can use to specify the style when creating a window. When set, the window will appear full screen. This mask is toggled when <code>toggleFullScreen:</code> is called.
<code>NSFullSizeContentViewWindowMask</code>	32768	One of the constants you can use to specify the style when creating a window. If set, the <code>contentView</code> will consume the full size of the window, combined with other window style masks, but is only respected when used with a titlebar. Utilizing this mask opts-in to layer-backing. Utilize the <code>contentView</code> 's <code>autoLayoutContentLayoutGuide</code> to layout views underneath the titlebar area.
<code>NSMiniaturizableWindowMask</code>	4	One of the constants you can use to specify the style when creating a window. The window displays a minimize button.
<code>NSResizableWindowMask</code>	8	One of the constants you can use to specify the style when creating a window. The window displays a resize control.
<code>NSSelectingNext</code>	1	One of the constants specify the direction a window is currently selecting the key view using <code>keyViewSelectionDirection</code> . The window is proceeding to the next valid key view.
<code>NSSelectingPrevious</code>	2	One of the constants specify the direction a window is currently selecting the key view using <code>keyViewSelectionDirection</code> . The window is proceeding to the previous valid key view.
<code>NSTexturedBackgroundWindowMask</code>	256	One of the constants you can use to specify the style when creating a window. The window displays with a metal-textured background. A window may be moved by clicking and dragging anywhere in the window's background. A bordered window with this mask gets rounded corners.
<code>NSTitledWindowMask</code>	1	One of the constants you can use to specify the style when creating a window. The window displays a title bar.
<code>NSUnifiedTitleAndToolbarWindowMask</code>	4096	One of the constants you can use to specify the style when creating a window.
<code>NSUnscaledWindowMask</code>	2048	One of the constants you can use to specify the style when creating a window.
<code>NSWindowBackingLocationDefault</code>	0	One of the constant to specify the window backing store location. Determined by the operating system. Available in Mac OS X v10.5 and later.
<code>NSWindowBackingLocationMainMemory</code>	2	One of the constant to specify the window backing store location. Physical memory. Available in Mac OS X v10.5 and later.
<code>NSWindowBackingLocationVideoMemory</code>	1	One of the constant to specify the window backing store location. Video memory. Available in Mac OS X v10.5 and later.
<code>NSWindowCloseButton</code>	0	One constants providing a way to access standard title bar buttons. The close button.
<code>NSWindowCollectionBehaviorCanJoinAllSpaces</code>	1	One of the constants for window collection behaviors related to Spaces. The window appears in all spaces. The menu bar behaves the same as in Spaces. Available in Mac OS X v10.5 and later.
<code>NSWindowCollectionBehaviorDefault</code>	0	One of the constants for window collection behaviors related to Spaces. The window can be associated to one space at a time. Available in Mac OS X v10.5 and later.
<code>NSWindowCollectionBehaviorFullScreenAuxiliary</code>	256	One of the constants for window collection behaviors. Windows with this collection behavior can be shown with the full screen auxiliary window.
<code>NSWindowCollectionBehaviorFullScreenPrimary</code>	128	One of the constants for window collection behaviors. Windows with this collection behavior can be shown with the full screen primary window.

Edge Constants

Constant	Value	Description
NSMaxXEdge	2	the maximum X edge. Typically right side.
NSMaxYEdge	3	The maximum Y edge. Topically the top edge of a window.
NSMinXEdge	0	the minimum X edge. Typically left side.
NSMinYEdge	1	Minimum Y. As coordinates are upside down in the Cocoa world, this is the bottom edge of a window.

Titlebar Separator Styles

Constant	Value	Description
NSTitlebarSeparatorStyleAutomatic	0	The titlebar separator style is determined by the window,Ãs configuration
NSTitlebarSeparatorStyleLine	2	The titlebar separator is a line.
NSTitlebarSeparatorStyleNone	1	There,Ãs no titlebar separator.
NSTitlebarSeparatorStyleShadow	3	The titlebar separator is a shadow.

Window Order Constants

Constant	Value	Description
NSWindowAbove	1	Moves the window above the indicated window.
NSWindowBelow	-1	Moves the window below the indicated window.
NSWindowOut	0	Moves the window off the screen.

Animation Behavior Constants.

Constant	Value	Description
NSWindowAnimationBehaviorAlertPanel	5	Default behavior for alert window.
NSWindowAnimationBehaviorDefault	0	let AppKit infer animation behavior for this window
NSWindowAnimationBehaviorDocumentWindow	3	Default behavior for document window.
NSWindowAnimationBehaviorNone	2	suppress inferred animations (don't animate)
NSWindowAnimationBehaviorUtilityWindow	4	Default behavior for utility window.

Tabbing Modes

Constant	Value	Description
NSWindowTabbingModeAutomatic	0	The system automatically prefers to tab this window when appropriate.
NSWindowTabbingModeDisallowed	2	The window explicitly should not prefer to tab when shown
NSWindowTabbingModePreferred	1	The window explicitly should prefer to tab when shown.

Title Visibility

Constant	Value	Description
<code>NSWindowTitleHidden</code>	1	The always hidden mode hides the title and moves the toolbar up into the area previously occupied by the title.
<code>NSWindowTitleHiddenWhenActive</code>	2	This mode hides the title when the window is active, and shows it when inactive. It does not adjust toolbars.
<code>NSWindowTitleVisible</code>	0	The default mode has a normal window title and titlebar buttons.

Toolbar Styles

Constant	Value	Description
<code>NSWindowToolbarStyleAutomatic</code>	0	The system displays the toolbar based on the window, the given configuration, and the user's preferences.
<code>NSWindowToolbarStyleExpanded</code>	1	The toolbar appears below the window title.
<code>NSWindowToolbarStylePreference</code>	2	The toolbar appears below the window title and items in the toolbar are centered and have increased margins.
<code>NSWindowToolbarStyleUnified</code>	3	The toolbar, when visible, appears inline with the window title.
<code>NSWindowToolbarStyleUnifiedCompact</code>	4	The toolbar appears inline and with the window title and with smaller margins when visible.

User Tabbing Preferences

Constant	Value	Description
<code>NSWindowUserTabbingPreferenceAlways</code>	1	The user always tabs windows.
<code>NSWindowUserTabbingPreferenceInFullScreen</code>	2	The user tabs windows only when in full screen.
<code>NSWindowUserTabbingPreferenceManual</code>	0	The user never tabs windows.

4.37 class NSWindowRestoreHandlerMBS

4.37.1 class NSWindowRestoreHandlerMBS

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: The class implementing the event to be called for window restoration.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 13.2pr1](#)

4.37.2 Methods

4.37.3 Constructor

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: The constructor.

4.37.4 Destructor

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: The destructor.

4.37.5 SetLastError(error as NSErrorMBS)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Sets the error to pass back from event.

4.37.6 SetRestoredWindow(win as DesktopWindow)

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

Function: Sets the window to pass back from event.

See also:

- [4.37.7 SetRestoredWindow\(win as NSWindowMBS\)](#) 546
- [4.37.8 SetRestoredWindow\(win as window\)](#) 546

4.37.7 SetRestoredWindow(win as NSWindowMBS)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Sets the window to pass back from event.

See also:

- 4.37.6 SetRestoredWindow(win as DesktopWindow) 545
- 4.37.8 SetRestoredWindow(win as window) 546

4.37.8 SetRestoredWindow(win as window)

Plugin Version: 13.2, Platform: macOS, Targets: Desktop only.

Function: Sets the window to pass back from event.

See also:

- 4.37.6 SetRestoredWindow(win as DesktopWindow) 545
- 4.37.7 SetRestoredWindow(win as NSWindowMBS) 546

4.37.9 Events

4.37.10 RestoreWindow(identifier as string, state as Variant)

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

Function: The following event is sent to request that a window be restored.

Notes: State is a NSCoderMBS object.

If the receiver knows how to restore the identified window, it should invoke the completion handler with the window, possibly creating it. It is acceptable to invoke the completion handler with a pre-existing window, though you should not pass the same window to more than one completion handler. If the receiver cannot restore the identified window (for example, the window referenced a document that has been deleted), it should invoke the completion handler with a nil window. In Mac OS X 10.7, the error parameter is ignored.

The receiver is passed the identifier of the window, which allows the receiver to quickly check for known windows. For example, you might give your preferences window an identifier of "preferences" in the nib, and then check for that identifier in your implementation. The receiver is also passed the NSCoder containing the combined restorable state of the window, its delegate, the window controller, and any document. The receiver may decode information previously stored in the coder to determine what window to restore.

Please call SetRestoredWindow or SetError to pass back status to system. It is not necessary or recommended for implementations of this method to order restored windows onscreen (for example, the window

4.37. *CLASS NSWINDOWRESTOREHANDLERMBS*

547

may have been minimized, in which case it will not be ordered onscreen).

Please call `setError` and `setRestoredWindow` to provide result to system.

Chapter 5

Cocoa Controls

5.1 class ContainerControl

5.1.1 class ContainerControl

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: The container class in Xojo.

5.1.2 Methods

5.1.3 NSViewMBS as NSViewMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Creates a NSViewMBS object for the given container.

Example:

```
MsgBox ContainerControl1.NSViewMBS.className
```

Notes: This way you can manipulate Cocoa container controls directly.

5.2 class DesktopContainer

5.2.1 class DesktopContainer

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

Function: The container class in Xojo.

5.2.2 Methods

5.2.3 NSViewMBS as NSViewMBS

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

Function: Creates a NSViewMBS object for the given container.

Example:

```
MsgBox ContainerControl1.NSViewMBS.className
```

Notes: This way you can manipulate Cocoa container controls directly.

Chapter 6

Cocoa Drawing

6.1 class NSBezierPathMBS

6.1.1 class NSBezierPathMBS

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The plugin class for a Cocoa NSBezierPath.

Notes: An NSBezierPath object allows you to create paths using PostScript-style commands. Paths consist of straight and curved line segments joined together. Paths can form recognizable shapes such as rectangles, ovals, arcs, and glyphs; they can also form complex polygons using either straight or curved line segments. A single path can be closed by connecting its two endpoints, or it can be left open.

An NSBezierPath object can contain multiple disconnected paths, whether they are closed or open. Each of these paths is referred to as a subpath. The subpaths of an NSBezierPath object must be manipulated as a group. The only way to manipulate subpaths individually is to create separate NSBezierPath objects for each.

For a given NSBezierPath object, you can stroke the path's outline or fill the region occupied by the path. You can also use the path as a clipping region for views or other regions. Using methods of NSBezierPath, you can also perform hit detection on the filled or stroked path. Hit detection is needed to implement interactive graphics, as in rubberbanding and dragging operations.

The current graphics context is automatically saved and restored for all drawing operations involving NSBezierPath objects, so your application does not need to worry about the graphics settings changing across invocations.

Blog Entries

- [MBS Real Studio Plugins, version 12.5pr4](#)

6.1.2 Methods

6.1.3 `appendBezierPath(path as NSBezierPathMBS)`

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

Function: Appends the contents of the specified path object to the receiver's path.

Notes: Path: The path to add to the receiver.

This method adds the commands used to create aPath to the end of the receiver's path. This method does not explicitly try to connect the subpaths in the two objects, although the operations in Path may still cause that effect.

6.1.4 `appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double)`

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

Function: Appends an arc of a circle to the receiver's path.

Notes: center: Specifies the center point of the circle used to define the arc.

radius: Specifies the radius of the circle used to define the arc.

startAngle: Specifies the starting angle of the arc, measured in degrees counterclockwise from the x-axis.

endAngle: Specifies the end angle of the arc, measured in degrees counterclockwise from the x-axis.

The created arc lies on the perimeter of the circle, between the angles specified by the startAngle and endAngle parameters. The arc is drawn in a counterclockwise direction. If the receiver's path is empty, this method sets the current point to the beginning of the arc before adding the arc segment. If the receiver's path is not empty, a line is drawn from the current point to the starting point of the arc.

Depending on the length of the arc, this method may add multiple connected curve segments to the path.

See also:

- 6.1.5 `appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean)` 552
- 6.1.6 `appendBezierPathWithArc(point1 as NSPointMBS, point2 as NSPointMBS, radius as Double)` 553

6.1.5 `appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean)`

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

Function: Appends an arc of a circle to the receiver's path.

Notes: center: Specifies the center point of the circle used to define the arc.

radius: Specifies the radius of the circle used to define the arc.

startAngle: Specifies the starting angle of the arc, measured in degrees counterclockwise from the x-axis.

endAngle: Specifies the end angle of the arc, measured in degrees counterclockwise from the x-axis.

clockwise: true if you want the arc to be drawn in a clockwise direction; otherwise false to draw the arc in a counterclockwise direction.

The created arc lies on the perimeter of the circle, between the angles specified by the startAngle and endAngle parameters. The arc is drawn in the direction indicated by the clockwise parameter. If the receiver's path is empty, this method sets the current point to the beginning of the arc before adding the arc segment. If the receiver's path is not empty, a line is drawn from the current point to the starting point of the arc.

Depending on the length of the arc, this method may add multiple connected curve segments to the path. See also:

- 6.1.4 appendBezierPathWithArc(center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double) 552
- 6.1.6 appendBezierPathWithArc(point1 as NSPointMBS, point2 as NSPointMBS, radius as Double) 553

6.1.6 appendBezierPathWithArc(point1 as NSPointMBS, point2 as NSPointMBS, radius as Double)

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

Function: Appends an arc to the receiver's path.

Notes: point1: The middle point of the angle.

point2: The end point of the angle.

radius: The radius of the circle inscribed in the angle.

The created arc is defined by a circle inscribed inside the angle specified by three points: the current point, the fromPoint parameter, and the toPoint parameter (in that order). The arc itself lies on the perimeter of the circle, whose radius is specified by the radius parameter. The arc is drawn between the two points of the circle that are tangent to the two legs of the angle.

The arc usually does not contain the points in the fromPoint and toPoint parameters. If the starting point of the arc does not coincide with the current point, a line is drawn between the two points. The starting point of the arc lies on the line defined by the current point and the fromPoint parameter.

You must set the path's current point (using the moveToPoint method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an NSGenericException exception.

Depending on the length of the arc, this method may add multiple connected curve segments to the path. See also:

- 6.1.4 `appendBezierPathWithArc:center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double` 552
- 6.1.5 `appendBezierPathWithArc:center as NSPointMBS, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean` 552

6.1.7 `appendBezierPathWithGlyph(glyph as Integer, font as NSFontMBS)`

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

Function: Appends an outline of the specified glyph to the receiver’s path.

Notes: Glyph: The glyph to add to the path.

font: The font in which the glyph is encoded.

If the glyph is not encoded in the font specified by the font parameter—that is, the font does not have an entry for the specified glyph—then no path is appended to the receiver.

You must set the path’s current point (using the `moveToPoint` method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an `NSGenericException` exception.

6.1.8 `appendBezierPathWithGlyphs(glyphs() as Integer, font as NSFontMBS)`

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

Function: Appends the outlines of the specified glyphs to the receiver’s path.

Notes: glyphs: An array of glyphs to add to the path.

count: The number of glyphs in the glyphs parameter.

font: The font in which the glyphs are encoded.

If the glyphs are not encoded in the font specified by the font parameter—that is, the font does not have an entry for one of the specified glyphs—then no path is appended to the receiver.

You must set the path’s current point (using the `moveToPoint` method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an `NSGenericException` exception.

6.1.9 appendBezierPathWithOvalInRect(rect as NSRectMBS)

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

Function: Appends an oval path to the receiver, inscribing the oval in the specified rectangle.

Notes: rect: The rectangle in which to inscribe the oval.

Before adding the oval, this method moves the current point, which implicitly closes the current subpath. If the aRect parameter specifies a square, the inscribed path is a circle. The path is constructed by starting in the lower-right quadrant of the rectangle and adding arc segments counterclockwise to complete the oval.

6.1.10 appendBezierPathWithPoints(points() as NSPointMBS)

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

Function: Appends a series of line segments to the receiver's path.

Notes: points: An array of NSPoint data types, each of which contains the end point of the next line segment.

count :The number of points in the points parameter.

This method interprets the points as a set of connected line segments. If the current path contains an open subpath, a line is created from the last point in that subpath to the first point in the points array. If the current path is empty, the first point in the points array is used to set the starting point of the line segments. Subsequent line segments are added using the remaining points in the array.

This method does not close the path that is created. If you wish to create a closed path, you must do so by explicitly invoking the receiver's closePath method.

6.1.11 appendBezierPathWithRect(rect as NSRectMBS)

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

Function: Appends a rectangular path to the receiver's path.

Notes: rect: The rectangle describing the path to create.

Before adding the rectangle, this method moves the current point to the origin of the rectangle, which implicitly closes the current subpath (if any). The path is constructed by starting at the origin of aRect and adding line segments in a counterclockwise direction. The final segment is added using a closePath message.

6.1.12 `appendBezierPathWithRoundedRect(rect as NSRectMBS, xRadius as Double, yRadius as Double)`

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

Function: Appends a rounded rectangular path to the receiver's path.

Notes: `rect`: The rectangle that defines the basic shape of the path.

`xRadius`: The radius of each corner oval along the x-axis. Values larger than half the rectangle's width are clamped to half the width.

`yRadius`: The radius of each corner oval along the y-axis. Values larger than half the rectangle's height are clamped to half the height.

The path is constructed in a counter-clockwise direction, starting at the top-left corner of the rectangle. If either one of the radius parameters contains the value 0.0, the returned path is a plain rectangle without rounded corners.

6.1.13 `bezierPath as NSBezierPathMBS`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Creates and returns a new `NSBezierPath` object.

6.1.14 `bezierPathByFlatteningPath as NSBezierPathMBS`

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

Function: Creates and returns a "flattened" copy of the receiver.

Notes: Returns a new path object whose contents are a flattened version of the receiver's path.

Flattening a path converts all curved line segments into straight line approximations. The granularity of the approximations is controlled by the path's current flatness value, which is set using the `DefaultFlatness` property.

6.1.15 `bezierPathByReversingPath as NSBezierPathMBS`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Creates and returns a new `NSBezierPath` object with the reversed contents of the receiver's path.

Notes: Returns a new path object whose contents are a reversed version of the receiver's path.

Reversing a path does not necessarily change the appearance of the path when rendered. Instead, it changes

the direction in which path segments are drawn. For example, reversing the path of a rectangle (whose line segments are normally drawn starting at the origin and proceeding in a counterclockwise direction) causes its line segments to be drawn in a clockwise direction instead. Drawing a reversed path could affect the appearance of a filled pattern, depending on the pattern and the fill rule in use.

This method reverses each whole or partial subpath in the path object individually.

6.1.16 `bezierPathWithOvalInRect(r as NSRectMBS) as NSBezierPathMBS`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Creates and returns a new NSBezierPath object initialized with an oval path inscribed in the specified rectangle.

Notes: r: The rectangle in which to inscribe an oval.

Returns an NSBezierPath new path object with the oval path.

If the rect parameter specifies a square, the inscribed path is a circle. The path is constructed by starting in the lower-right quadrant of the rectangle and adding arc segments counterclockwise to complete the oval.

6.1.17 `bezierPathWithRect(r as NSRectMBS) as NSBezierPathMBS`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Creates and returns a new NSBezierPath object initialized with a rectangular path.

Notes: r: The rectangle describing the path to create.

Returns a new path object with the rectangular path.

The path is constructed by starting at the origin of aRect and adding line segments in a counterclockwise direction.

6.1.18 `bezierPathWithRoundedRect(r as NSRectMBS, xRadius as Double, yRadius as Double) as NSBezierPathMBS`

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

Function: Creates and returns a new NSBezierPath object initialized with a rounded rectangular path.

Notes: r: The rectangle that defines the basic shape of the path.

xRadius: The radius of each corner oval along the x-axis. Values larger than half the rectangle's width are clamped to half the width.

`yRadius`: The radius of each corner oval along the y-axis. Values larger than half the rectangle's height are clamped to half the height.

Returns a new path object with the rounded rectangular path.

The path is constructed in a counter-clockwise direction, starting at the top-left corner of the rectangle. If either one of the radius parameters contains the value 0.0, the returned path is a plain rectangle without rounded corners.

6.1.19 `closePath`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Closes the most recently added subpath.

Notes: This method closes the current subpath by creating a line segment between the first and last points in the subpath. This method subsequently updates the current point to the end of the newly created line segment, which is also the first point in the now closed subpath.

6.1.20 `Constructor`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Initializes a new empty `NSBezierPath` object.

6.1.21 `containsPoint(p as NSPointMBS) as boolean`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the receiver contains the specified point.

Example:

```
dim r as new NSRectMBS(10, 10, 50, 50)
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPathWithRect(r)
dim p as new NSPointMBS(20,20)
```

```
MsgBox str(b.containsPoint(p))
```

Notes: `p`: The point to test against the path, specified in the path object's coordinate system.

Returns true if the path's enclosed area contains the specified point; otherwise, false.

This method checks the point against the path itself and the area it encloses. When determining hits in the enclosed area, this method uses the non-zero winding rule (`NSNonZeroWindingRule`). It does not take into account the line width used to stroke the path.

6.1.22 copy as NSBezierPathMBS

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Creates a copy of the bezier path.

6.1.23 curveToPoint(endPoint as NSPointMBS, controlPoint1 as NSPointMBS, controlPoint2 as NSPointMBS)

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

Function: Adds a Bezier cubic curve to the receiver's path.

Notes: `endPoint`: The destination point of the curve segment, specified in the current coordinate system

`controlPoint1`: The point that determines the shape of the curve near the current point.

`controlPoint2`: The point that determines the shape of the curve near the destination point.

You must set the path's current point (using the `moveToPoint` method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an `NSGenericException` exception.

6.1.24 elementAtIndex(index as Integer) as Integer

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

Function: Returns the type of path element at the specified index.

Notes: `index`: The index of the desired path element.

Returns the type of the path element.

Path elements describe the commands used to define a path and include basic commands such as moving to a specific point, creating a line segment, creating a curve, or closing the path. The elements are stored in the order of their execution.

See also:

- 6.1.25 `elementAtIndex(index as Integer, byref associatedPoints() as NSPointMBS) as Integer` 560

6.1.25 `elementAtIndex(index as Integer, byref associatedPoints() as NSPointMBS) as Integer`

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

Function: Gets the element type and (and optionally) the associated points for the path element at the specified index.

Notes: index: The index of the desired path element.

associatedPoints: On output, the data points associated with the specified path element.

Returns the type of the path element.

For curve operations, the order of the points is controlPoint1 (associatedPoints(0)), controlPoint2 (associatedPoints(1)), endPoint (associatedPoints(2)).

See also:

- 6.1.24 `elementAtIndex(index as Integer) as Integer`

559

6.1.26 `elementCount as Integer`

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

Function: Returns the total number of path elements in the receiver's path.

Example:

```
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath
```

```
b.moveToPoint NSMakePointMBS(10, 10)
b.lineToPoint NSMakePointMBS(290, 290)
MsgBox str(b.elementCount) // 2
```

```
b.removeAllPoints
MsgBox str(b.elementCount) // 0
```

Notes: Each element type corresponds to one of the operations described in constants.

6.1.27 `getLineDash(byref pattern() as Double, byref count as Integer, byref phase as Double)`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Returns the line-stroking pattern for the receiver.

Notes: pattern: On output, this array contains the lengths (measured in points) of the line segments and

gaps in the pattern. The values in the array alternate, starting with the first line segment length, followed by the first gap length, followed by the second line segment length, and so on.

count: On output, the number of entries written to pattern.

phase: On output, this value contains the offset at which to start drawing the pattern, measured in points along the dashed-line pattern. For example, a phase of 6 in the pattern 5-2-3-2 would cause drawing to begin in the middle of the first gap.

6.1.28 isEmpty as boolean

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the receiver is empty.

Example:

```
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath
MsgBox "isEmpty: "+str(b.isEmpty)
```

Notes: True if the receiver contains no path elements; otherwise, false.

6.1.29 lineToPoint(p as NSPointMBS)

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

Function: Appends a straight line to the receiver's path.

Notes: p: The destination point of the line segment, specified in the current coordinate system.

This method creates a straight line segment starting at the current point and ending at the point specified by the aPoint parameter. The current point is the last point in the receiver's most recently added segment.

You must set the path's current point (using the moveToPoint method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an NSGenericException exception.

6.1.30 moveToPoint(p as NSPointMBS)

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Moves the receiver's current point to the specified location.

Notes: p: A point in the current coordinate system.

This method implicitly closes the current subpath (if any) and sets the current point to the value in `p`. When closing the previous subpath, this method does not cause a line to be created from the first and last points in the subpath.

For many path operations, you must invoke this method before issuing any commands that cause a line or curve segment to be drawn.

6.1.31 `relativeCurveToPoint(endPoint as NSPointMBS, controlPoint1 as NSPointMBS, controlPoint2 as NSPointMBS)`

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

Function: Adds a Bezier cubic curve to the receiver's path from the current point to a new location, which is specified as a relative distance from the current point.

Notes: `endPoint`: The destination point of the curve segment, interpreted as a relative offset from the current point.

`controlPoint1`: The point that determines the shape of the curve near the current point, interpreted as a relative offset from the current point.

`controlPoint2`: The point that determines the shape of the curve near the destination point, interpreted as a relative offset from the current point.

You must set the path's current point (using the `moveToPoint` method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an `NSGenericException` exception.

6.1.32 `relativeLineToPoint(p as NSPointMBS)`

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

Function: Appends a straight line segment to the receiver's path starting at the current point and moving towards the specified point, relative to the current location.

Notes: `p`: A point whose coordinates are interpreted as a relative offset from the current point.

The destination point is relative to the current point. For example, if the current point is (1, 1) and `aPoint` contains the value (1, 2), a line segment is created between the points (1, 1) and (2, 3).

You must set the path's current point (using the `moveToPoint` method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an `NSGenericException` exception.

6.1.33 relativeMoveToPoint(p as NSPointMBS)

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

Function: Moves the receiver's current point to a new point whose location is the specified distance from the current point.

Notes: p: A point whose coordinates are interpreted as a relative offset from the current point.

This method implicitly closes the current subpath (if any) and updates the location of the current point. For example, if the current point is (1, 1) and aPoint contains the value (1, 2), the previous subpath would be closed and the current point would become (2, 3). When closing the previous subpath, this method does not cause a line to be created from the first and last points in the subpath.

You must set the path's current point (using the moveToPoint method or through the creation of a preceding line or curve segment) before you invoke this method. If the path is empty, this method raises an NSRangeException exception.

6.1.34 removeAllPoints

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Removes all path elements from the receiver, effectively clearing the path.

Example:

```
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath
```

```
b.moveToPoint NSMakePointMBS(10, 10)
b.lineToPoint NSMakePointMBS(290, 290)
MsgBox str(b.elementCount) // 2
```

```
b.removeAllPoints
MsgBox str(b.elementCount) // 0
```

6.1.35 setAssociatedPoints(points() as NSPointMBS, index as Integer)

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

Function: Changes the points associated with the specified path element.

Notes: points: An array containing up to three NSPointMBS data types. This parameter must contain the correct number of points for the path element at the specified index. Move, close path, and line segment commands require one point. Curve operations require three points.

index: The index of the path element you want to modify.

You can use this method to change the points associated with a path quickly and without recreating the path. You cannot use this method to change the type of the path element.

The following example shows you how you would modify the point associated with a line path element. The path created by this example results in a path with two elements. The first path element specifies a move to point (0, 0) while the second creates a line to point (100, 100). It then changes the line to go only to the point (50,50) using this method:

Note: If you specify too few points for a path element of type `NSCurveToBezierPathElement`, the behavior of this method is undefined.

6.1.36 `setLineDash(pattern() as Double, phase as Double)`

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Sets the line-stroking pattern for the receiver.

Example:

```
dim n as new NSImageMBS(300, 300)
dim g as new NSGraphicsMBS(n)

g.setFill color NSColorMBS.redColor

dim r as NSRectMBS = NSMakeRectMBS(50, 50, 100, 100)
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath

b.moveToPoint NSMakePointMBS(10, 10)
b.lineToPoint NSMakePointMBS(290, 290)

dim pattern() as Double
pattern.Append 5
pattern.Append 2
b.setLineDash pattern, 6

g.stroke(b)

g = nil

window1.Backdrop = n.CopyPicture // red line with dash pattern
```

Notes: `pattern`: An array of floating point values that contains the lengths (measured in points) of the line segments and gaps in the pattern. The values in the array alternate, starting with the first line segment length, followed by the first gap length, followed by the second line segment length, and so on
`count`: The number of values in `pattern`.

phase: The offset at which to start drawing the pattern, measured in points along the dashed-line pattern. For example, a phase of 6 in the pattern 5-2-3-2 would cause drawing to begin in the middle of the first gap.

6.1.37 transformUsingAffineTransform(transform as Variant)

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

Function: Transforms all points in the receiver using the specified transform.

Example:

```
dim bezierPath as NSBezierPathMBS = NSBezierPathMBS.bezierPath
dim transform as NSAffineTransformMBS = NSAffineTransformMBS.transform
```

```
bezierPath.moveToPoint NSMakePointMBS(0.0, 0.0)
bezierPath.lineToPoint NSMakePointMBS(100.0, 100.0)
```

```
transform.translate(10,10)
bezierPath.transformUsingAffineTransform(transform)
```

Break // bezierPath.bounds starts now at 10/10

Notes: Transform: The transform to apply to the path.

This method applies the transform to the path's points immediately. The following code translates a line from 0,0 to 100,100 to a line from 10,10 to 110,110.

transform must be a NSAffineTransformMBS object.

6.1.38 Properties

6.1.39 Bounds as NSRectMBS

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Returns the bounding box of the receiver's path.

Example:

```
dim r as new NSRectMBS(10, 10, 50, 50)
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPathWithRect(r)
dim rr as NSRectMBS = b.Bounds
MsgBox rr.String
```

Notes: The rectangle that encloses the path of the receiver. If the path contains curve segments, the bounding box encloses the curve but may not enclose the control points used to calculate the curve.
(Read only property)

6.1.40 ControlPointBounds as NSRectMBS

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

Function: Returns the bounding box of the receiver's path, including any control points.

Notes: The rectangle that encloses the receiver's path. If the path contains curve segments, the bounding box encloses the control points of the curves as well as the curves themselves.
(Read only property)

6.1.41 CurrentPoint as NSPointMBS

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: Returns the receiver's current point (the trailing point or ending point in the most recently added segment).

Example:

```
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath
b.moveToPoint NSMakePointMBS(10,10)
```

```
MsgBox b.CurrentPoint.String
```

Notes: The point from which the next drawn line or curve segment begins. If the receiver is empty, this method raises `NSGenericException`.
(Read only property)

6.1.42 defaultFlatness as Double

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

Function: The default flatness value for all paths.

Notes: The flatness value specifies the accuracy (or smoothness) with which curves are rendered. It is also the maximum error tolerance (measured in pixels) for rendering curves, where smaller numbers give smoother curves at the expense of more computation. The exact interpretation may vary slightly on different rendering devices.

The default flatness value is 0.6, which yields smooth curves.
(Read and Write property)

6.1.43 defaultLineCapStyle as Integer

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

Function: The default line cap style for all paths.

Notes: The default line cap style or NSButtLineCapStyle if no other style has been set. For a list of values, see Constants.

The default line cap style can be overridden for individual paths by setting a custom style for that path using the LineCapStyle property.
(Read and Write property)

6.1.44 defaultLineJoinStyle as Integer

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

Function: The default line join style for all paths.

Notes: The default line join style or NSMiterLineJoinStyle if no other value has been set. For a list of values, see Constants.

(Read and Write property)

6.1.45 defaultLineWidth as Double

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

Function: Returns the default line width for the all paths.

Notes: The line width defines the thickness of stroked paths. A width of 0 is interpreted as the thinnest line that can be rendered on a particular device. The actual rendered line width may vary from the specified width by as much as 2 device pixels, depending on the position of the line with respect to the pixel grid and the current anti-aliasing settings. The width of the line may also be affected by scaling factors specified in the current transformation matrix of the active graphics context.

(Read and Write property)

6.1.46 defaultMiterLimit as Double

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

Function: Returns the default miter limit for all paths.

Notes: The miter limit helps you avoid spikes at the junction of two line segments connected by a miter join (`NSMiterLineJoinStyle`). If the ratio of the miter length—the diagonal length of the miter join—to the line thickness exceeds the miter limit, the joint is converted to a bevel join. The default miter limit value is 10, which converts miters whose angle at the joint is less than 11 degrees.

(Read and Write property)

6.1.47 `defaultWindingRule` as Integer

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

Function: Returns the default winding rule used to fill all paths.

Notes: The current default winding rule or `NSNonZeroWindingRule` if no default rule has been set. This value may be either `NSNonZeroWindingRule` or `NSEvenOddWindingRule`.

Winding rules determine how to paint (or fill) the region enclosed by a path. You use this method to set the default rule that is applied to paths that do not have a custom winding rule assigned.

(Read and Write property)

6.1.48 `Handle` as Integer

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

6.1.49 `flatness` as Double

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The flatness value of the path.

Notes: If no value is set, this method returns the default flatness value.

(Read and Write computed property)

6.1.50 `lineCapStyle` as Integer

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The line cap style for the receiver's path.

Notes: If this value is not set for the receiver, the default line cap style is returned.

(Read and Write computed property)

6.1.51 lineJoinStyle as Integer

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The receiver's line join style.

Notes: If this value is not set for the receiver, the default line join style is returned.

(Read and Write computed property)

6.1.52 lineWidth as Double

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The line width of the receiver's path.

Example:

```
dim n as new NSImageMBS(300, 300)
dim g as new NSGraphicsMBS(n)

g.setFill-color NSColorMBS.redColor

dim r as NSRectMBS = NSMakeRectMBS(50, 50, 100, 100)
dim b as NSBezierPathMBS = NSBezierPathMBS.bezierPath

b.moveToPoint NSMakePointMBS(10, 10)
b.lineToPoint NSMakePointMBS(290, 290)

dim pattern() as Double
pattern.Append 10
pattern.Append 4

b.lineWidth = 5
b.setLineDash pattern, 12

g.stroke(b)

g = nil

window1.Backdrop = n.CopyPicture // red line with dash pattern
```

Notes: The line width of the receiver, measured in points in the user coordinate space. If no value was set explicitly for the receiver, this method returns the default line width.

(Read and Write computed property)

6.1.53 miterLimit as Double

Plugin Version: 12.5, Platform: macOS, Targets: All.

Function: The miter limit of the receiver's path.

Notes: If no value is set, this method returns the default miter limit.

(Read and Write computed property)

6.1.54 windingRule as Integer

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

Function: Returns the winding rule used to fill the receiver's path.

Notes: Returns the winding rule for the path. This value may be either `NSNonZeroWindingRule` or `NSEven-OddWindingRule`.

This value overrides the default value returned by `defaultWindingRule`.

For more information on how winding rules affect the appearance of filled paths, see Constants.

(Read and Write computed property)

6.1.55 Constants

Line Joint Style Constants

Constant	Value	Description
<code>NSBevelLineJoinStyle</code>	2	Specifies a bevel line shape of the joints between connected segments of a stroked path. See the <code>setDefaultLineJoinStyle</code> method for an example of the appearance.
<code>NSMiterLineJoinStyle</code>	0	Specifies a miter line shape of the joints between connected segments of a stroked path. See the <code>setDefaultLineJoinStyle</code> method for an example of the appearance.
<code>NSRoundLineJoinStyle</code>	1	Specifies a round line shape of the joints between connected segments of a stroked path. See the <code>setDefaultLineJoinStyle</code> method for an example of the appearance.

Line Cap Style Constants

Constant	Value	Description
NSButtLineCapStyle	0	Specifies a butt line cap style for endpoints for an open path when stroked. See the <code>setDefaultLineCapStyle</code> method for an example of the appearance.
NSRoundLineCapStyle	1	Specifies a round line cap style for endpoints for an open path when stroked. See the <code>setDefaultLineCapStyle</code> method for an example of the appearance.
NSSquareLineCapStyle	2	Specifies a square line cap style for endpoints for an open path when stroked. See the <code>setDefaultLineCapStyle</code> method for an example of the appearance.

Path Element Command Constants

Constant	Value	Description
NSClosePathBezierPathElement	3	Marks the end of the current subpath at the specified point. Note that the point specified for the Close Path element is essentially the same as the current point.
NSCurveToBezierPathElement	2	Creates a curved line segment from the current point to the specified endpoint using two control points to define the curve. The points are stored in the following order: <code>controlPoint1</code> , <code>controlPoint2</code> , <code>endPoint</code> . Ovals, arcs, and Bezier curves all use curve elements to specify their geometry. Contains 3 points.
NSLineToBezierPathElement	1	Creates a straight line from the current drawing point to the specified point. Lines and rectangles are specified using this path element. Contains 1 point.
NSMoveToBezierPathElement	0	Moves the path object's current drawing point to the specified point. This path element does not result in any drawing. Using this command in the middle of a path results in a disconnected line segment. Contains 1 point.

Window Rule Constants

Constant	Value	Description
NSEvenOddWindingRule	1	Specifies the even-odd winding rule. Count the total number of path crossings. If the number of crossings is even, the point is outside the path. If the number of crossings is odd, the point is inside the path and the region containing it should be filled.
NSNonZeroWindingRule	0	Specifies the non-zero winding rule. Count each left-to-right path as +1 and each right-to-left path as -1. If the sum of all crossings is 0, the point is outside the path. If the sum is nonzero, the point is inside the path and the region containing it is filled. This is the default winding rule.

6.2 class NSBitmapImageRepMBS

6.2.1 class NSBitmapImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSBitmapImageRep class renders an image from bitmap data.

Notes: Bitmap data formats supported include GIF, JPEG, TIFF, PNG, and various permutations of raw bitmap data.

Alpha Premultiplication

If a coverage (alpha) plane exists, a bitmap's color components are premultiplied with it. If you modify the contents of the bitmap, you are therefore responsible for premultiplying the data. For this reason, though, if you want to manipulate the actual data, an NSBitmapImageRep object is not recommended for storage. If you need to work with data that is not premultiplied, you should use Quartz, specifically CGImageCreate with kCGImageAlphaLast.

Note that premultiplying does not affect the output quality. Given source bitmap pixel s , destination pixel d , and alpha value a , a blend is basically

$$d' = a * s + (1 - a) * d$$

All premultiplication does is precalculate $a * s$.

Subclass of the NSImageRepMBS class.

Blog Entries

- [MBS Xojo Plugins, version 17.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr3](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr5](#)
- [MBS Real Studio Plugins, version 12.3pr7](#)
- [MBS Real Studio Plugins, version 12.0pr5](#)

6.2.2 Methods

6.2.3 bitmapImageRepByConvertingToColorSpace(colorSpace as NSColorSpaceMBS, renderingIntent as Integer) as NSBitmapImageRepMBS

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Converts the image rep to the specified colorspace.

Notes: targetSpace: The new colorSpace

renderingIntent: The rendering intent specifies how to handle colors that are not located within the target color space. The supported values are NSColorRenderingIntent.

Constant	Value
Default	0
AbsoluteColorimetric	1
RelativeColorimetric	2
Perceptual	3
Saturation	4

An NSBitmapImageRep, or nil, if the conversion fails. If the original NSBitmapImageRep already uses that colorSpace, it is returned as is.

6.2.4 bitmapImageRepByRetaggingWithColorSpace(newSpace as NSColorSpaceMBS) as NSBitmapImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Changes the colorSpace tag of the receiver.

Notes: newSpace: The desired colorSpace.

Returns an NSBitmapImageRep, or nil, if the conversion fails. If the original NSBitmapImageRep already uses that colorSpace, it is returned as is.

This method will definitely fail if you pass a colorSpace that has a different color space model than the receiver. That is, if your original image is sRGB, you can only retag with some other RGB colorspace. Available in Mac OS X v10.6 and later.

6.2.5 BMPRepresentation(properties as dictionary = nil) as Memoryblock

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

Function: Returns the bitmap as bmp data.

6.2.6 canBeCompressedUsing(compression as Integer) as Boolean

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tests whether the receiver can be compressed by the specified compression scheme.

Notes: Returns true if the receiver's data matches compression with this type, false if the data doesn't match compression or if compression is unsupported..

Legal values for compression can be found in NSBitmapImageRep.h and are described in TIFF Compression in NSBitmapImageReps. This method returns true if the receiver's data matches compression; for example, if compression is NSTIFFCompressionCCITTFAX3, then the data must be 1 bit per sample and 1 sample per pixel.

6.2.7 Constructor(data as Memoryblock)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated NSBitmapImageRep from the provided data.

Notes: data: A data object containing image data. The contents of bitmapData can be any supported bitmap format. For TIFF data, the NSBitmapImageRep is initialized from the first header and image data found in bitmapData.

On success the handle property is not zero.

See also:

- 6.2.8 Constructor(pic as Picture) 574
- 6.2.9 Constructor(pixelsWide as Integer, pixelsHigh as Integer, bitsPerSample as Integer, samplesPerPixel as Integer, hasAlpha as boolean, colorSpaceName as string, bytesPerRow as Integer, bitsPerPixel as Integer) 575

6.2.8 Constructor(pic as Picture)

Plugin Version: 14.1, Platform: macOS, Targets: All.

Function: Creates an NSBitmapImageRep object created from a Core Graphics image object.

Notes: pic: A Core Graphics image object (an opaque type) from which to create the receiver. This opaque type is retained.

An NSBitmapImageRep object initialized from the contents of the Core Graphics image or nil if the NSBitmapImageRep couldn't be created.

If you use this method, you should treat the resulting bitmap NSBitmapImageRep object as read only. Because it only retains the value in the cgImage parameter, rather than unpacking the data, accessing the pixel data requires the creation of a copy of that data in memory. Changes to that data are not saved back to the Core Graphics image.

Available in OS X v10.5 and later.

See also:

- 6.2.7 Constructor(data as Memoryblock) 574
- 6.2.9 Constructor(pixelsWide as Integer, pixelsHigh as Integer, bitsPerSample as Integer, samplesPerPixel as Integer, hasAlpha as boolean, colorSpaceName as string, bytesPerRow as Integer, bitsPerPixel as Integer) 575

6.2.9 Constructor(pixelsWide as Integer, pixelsHigh as Integer, bitsPerSample as Integer, samplesPerPixel as Integer, hasAlpha as boolean, colorSpaceName as string, bytesPerRow as Integer, bitsPerPixel as Integer)

Plugin Version: 12.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes a newly allocated NSBitmapImageRep object, so it can render the specified image.

Example:

```
dim n as new NSImageMBS(300,300)
dim r as new NSBitmapImageRepMBS(300, 300, 8, 4, true, NSColorSpaceMBS.NSCalibratedRGBColorSpace, 4*300, 32)
```

```
dim g as new NSGraphicsMBS(r)
g.SetColorRGB 1.0,0,0,0.5
g.fillRect 0, 0, 100, 100
g = nil // flush
n.addRepresentation r
```

```
Backdrop = n.CopyPictureWithMask
```

Notes: The image bitmap data is allocated for you.

pixelsWide: The width of the image in pixels. This value must be greater than 0.

pixelsHigh: The height of the image in pixels. This value must be greater than 0.

bitsPerSample: The number of bits used to specify one pixel in a single component of the data. All components are assumed to have the same bits per sample. bps should be one of these values: 1, 2, 4, 8, 12, or 16.

samplesPerPixel: The number of data components, or samples per pixel. This value includes both color components and the coverage component (alpha), if present. Meaningful values range from 1 through 5. An image with cyan, magenta, yellow, and black (CMYK) color components plus a coverage component would have an spp of 5; a grayscale image that lacks a coverage component would have an spp of 1.

hasAlpha: True if one of the components counted in the number of samples per pixel (spp) is a coverage (alpha) component, and false if there is no coverage component. If true, the color components in the bitmap data must be premultiplied with their coverage component.

colorSpaceName: A string constant that indicates how data values are to be interpreted. It should be one of the following values: NSCalibratedWhiteColorSpace, NSCalibratedBlackColorSpace, NSCalibrate-

dRGBColorSpace, NSDeviceWhiteColorSpace, NSDeviceBlackColorSpace, NSDeviceRGBColorSpace, NSDeviceCMYKColorSpace, NSNamedColorSpace or NSCustomColorSpace. (see NSColorSpaceMBS)

If bps is 12, you cannot specify the monochrome color space.

bytesPerRow: The number of bytes that are allocated for each scan line in each plane of data. A scan line is a single row of pixels spanning the width of the image.

Normally, rowBytes can be figured from the width of the image, the number of bits per pixel in each sample (bps), and, if the data is in a meshed configuration, the number of samples per pixel (spp). However, if the data for each row is aligned on word or other boundaries, it may have been necessary to allocate more memory for each row than there is data to fill it. rowBytes lets the object know whether that's the case.

If you pass in a rowBytes value of 0, the bitmap data allocated may be padded to fall on long word or larger boundaries for performance. If your code wants to advance row by row, use bytesPerRow and do not assume the data is packed. Passing in a non-zero value allows you to specify exact row advances.

bitsPerPixel: This integer value informs NSBitmapImageRep how many bits are actually allocated per pixel in each plane of data. If the data is in planar configuration, this normally equals bps (bits per sample). If the data is in meshed configuration, it normally equals bps times spp (samples per pixel). However, it's possible for a pixel specification to be followed by some meaningless bits (empty space), as may happen, for example, if pixel data is aligned on byte boundaries. NSBitmapImageRep supports only a limited number of pixelBits values (other than the default): for RGB images with 4 bps, pixelBits may be 16; for RGB images with 8 bps, pixelBits may be 32. The legal values for pixelBits are system dependent.

If you specify 0 for this parameter, the object interprets the number of bits per pixel using the values in the bps and spp parameters, as described in the preceding paragraph, without any meaningless bits.

On success the handle property is not zero.

You can add several NSBitmapImageRepMBS to NSImage to make a multi page tiff file.

To make a 300 dpi image, the size of the NSImage must be smaller than the size of the bitmap inside by factor 72/300.

See also:

- 6.2.7 Constructor(data as Memoryblock) 574
- 6.2.8 Constructor(pic as Picture) 574

6.2.10 GIFRepresentation(properties as dictionary = nil) as Memoryblock

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

Function: Returns the bitmap as gif data.

6.2.11 imageRepWithCGImage(CGImage as Variant) as NSBitmapImageRepMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates an NSBitmapImageRep object created from a Core Graphics image object.

Notes: cgimage: A Core Graphics image object (an opaque type) from which to create the receiver. This

opaque type is retained.

An NSBitmapImageRep object initialized from the contents of the Core Graphics image or nil if the NSBitmapImageRep couldn't be created.

If you use this method, you should treat the resulting bitmap NSBitmapImageRep object as read only. Because it only retains the value in the cgImage parameter, rather than unpacking the data, accessing the pixel data requires the creation of a copy of that data in memory. Changes to that data are not saved back to the Core Graphics image.

Available in OS X v10.5 and later.

6.2.12 imageRepWithCGImageRef(CGImageHandle as Integer) as NSBitmapImageRepMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates an NSBitmapImageRep object created from a Core Graphics image reference.

Notes: CGImageHandle: A Core Graphics image object (an opaque type) from which to create the receiver. This opaque type is retained.

An NSBitmapImageRep object initialized from the contents of the Core Graphics image or nil if the NSBitmapImageRep couldn't be created.

If you use this method, you should treat the resulting bitmap NSBitmapImageRep object as read only. Because it only retains the value in the cgImage parameter, rather than unpacking the data, accessing the pixel data requires the creation of a copy of that data in memory. Changes to that data are not saved back to the Core Graphics image.

Available in OS X v10.5 and later.

6.2.13 imageRepWithCIImage(CIImage as variant) as NSBitmapImageRepMBS

Plugin Version: 17.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates an NSBitmapImageRep object created from a CoreImage object.

Notes: CIImage: A Core Graphics image object (an opaque type) from which to create the receiver. This opaque type is retained.

An NSBitmapImageRep object initialized from the contents of the Core Graphics image or nil if the NS-

BitmapImageRep couldn't be created.

If you use this method, you should treat the resulting bitmap NSBitmapImageRep object as read only. Because it only retains the value in the CIImage parameter, rather than unpacking the data, accessing the pixel data requires the creation of a copy of that data in memory. Changes to that data are not saved back to the Core Graphics image.

Available in OS X v10.5 and later.

6.2.14 imageRepWithCIImageRef(CIImageHandle as Integer) as NSBitmapImageRepMBS

Plugin Version: 17.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates an NSBitmapImageRep object created from a CoreImage reference.

Notes: CIImageHandle: A Core Graphics image object (an opaque type) from which to create the receiver. This opaque type is retained.

An NSBitmapImageRep object initialized from the contents of the Core Graphics image or nil if the NSBitmapImageRep couldn't be created.

If you use this method, you should treat the resulting bitmap NSBitmapImageRep object as read only. Because it only retains the value in the CIImage parameter, rather than unpacking the data, accessing the pixel data requires the creation of a copy of that data in memory. Changes to that data are not saved back to the Core Graphics image.

Available in OS X v10.5 and later.

6.2.15 imageRepWithData(data as Memoryblock) as NSBitmapImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSBitmapImageRep object initialized with the first image in the supplied data.

Notes: data: A data object containing one or more bitmapped images. The bitmapData parameter can contain data in any supported bitmap format.

Returns an NSBitmapImageRep instance or nil if the class is unable to create an image representation.

6.2.16 JPEGRepresentation(properties as dictionary = nil) as Memoryblock

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

Function: Returns the bitmap as jpeg data.

6.2.17 NSImageColorSyncProfileData as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a memoryblock containing the ColorSync profile data.

It can be used for TIFF, JPEG, GIF, and PNG files. This value is set when reading in and used when writing out image data. You can get the profile data for a particular color space from the corresponding NSColorSpace object or from the ColorSync Manager.

6.2.18 NSImageCompressionFactor as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the compression factor of the image.

Used only for JPEG files. JPEG compression in TIFF files is not supported, and the factor is ignored. The value is a float between 0.0 and 1.0, with 1.0 resulting in no compression and 0.0 resulting in the maximum compression possible. It's set when reading in and used when writing out the image.

6.2.19 NSImageCompressionMethod as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number identifying the compression method of the image.

Used only for TIFF files. The value corresponds to one of the NSTIFFCompression constants, described below. It's set when reading in and used when writing out.

6.2.20 NSImageCurrentFrame as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the current frame for an animated GIF file.

The first frame is 0.

6.2.21 `NSImageCurrentFrameDuration` as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the duration (in seconds) of the current frame for an animated GIF image.

The frame duration can be a floating-point value. It is used when reading in, but not when writing out.

6.2.22 `NSImageDitherTransparency` as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing a boolean that indicates whether the image is dithered.

Used only when writing GIF files.

6.2.23 `NSImageEXIFData` as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies an dictionary containing the EXIF data for the image.

This property is used only when reading or writing JPEG files. The dictionary contains the EXIF keys and values. The standard dictionary keys (that is, those that are not specific to camera vendors) are identical to those for `kCGImagePropertyExifDictionary` declared in the `CGImageSource` API. See `kCGImagePropertyExifDictionary` Keys for details.

6.2.24 `NSImageFallbackBackgroundColor` as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Specifies the background color to use when writing to an image format (such as JPEG) that doesn't support alpha. The color's alpha value is ignored. The default background color, when this property is not specified, is white. The value of the property should be an `NSColorMBS` object. This constant corresponds to the `kCGImageDestinationBackgroundColor` constant in Quartz.

Available in Mac OS X v10.5 and later.

6.2.25 NSImageFrameCount as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the number of frames in an animated GIF file. This value is used when reading in data.

6.2.26 NSImageGamma as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the gamma value for the image.

Used only for PNG files. The gamma values is a floating-point number between 0.0 and 1.0, with 0.0 being black and 1.0 being the maximum color. It's set when reading in and used when writing out.

6.2.27 NSImageInterlaced as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing a Boolean value that indicates whether the image is interlaced. Used only when writing out PNG files.

6.2.28 NSImageLoopCount as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing the number of loops to make when animating a GIF image.

A value of 0 indicates the animation should loop indefinitely. Values should be specified as Integer numbers. It is used when reading in but not when writing out the image.

6.2.29 NSImageProgressive as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies a number containing a boolean that indicates whether the image uses progressive encoding. Used only for JPEG files. It's set when reading in and used when writing out.

6.2.30 NSImageRGBColorTable as string

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the property keys.

Notes: Identifies an Memoryblock containing the RGB color table.

Used only for GIF files. It's stored as packed RGB. It's set when reading in and used when writing out.

6.2.31 PNGRepresentation(properties as dictionary = nil) as Memoryblock

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

Function: Returns the bitmap as PNG data.

6.2.32 TIFFRepresentation as Memoryblock

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a TIFF representation of the receiver.

Notes: This method invokes `TIFFRepresentationUsingCompression` using the stored compression type and factor retrieved from the initial image data or changed using `setCompression`. If the stored compression type isn't supported for writing TIFF data (for example, `NSTIFFCompressionNEXT`), the stored compression is changed to `NSTIFFCompressionNone` before invoking `TIFFRepresentationUsingCompression`. receiver, using the compression that's returned by `getCompression` (if applicable).

If a problem is encountered during generation of the TIFF, `TIFFRepresentation` raises an `NSTIFFException` or an `NSBadBitmapParametersException`.

See also:

- 6.2.33 `TIFFRepresentation(properties as dictionary = nil) as Memoryblock` 582

6.2.33 TIFFRepresentation(properties as dictionary = nil) as Memoryblock

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

Function: Returns the bitmap as tiff data.

See also:

- 6.2.32 `TIFFRepresentation as Memoryblock` 582

6.2.34 Properties

6.2.35 bitmapData as Ptr

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a pointer to the bitmap data.

Notes: If the data is planar, returns a pointer to the first plane.

(Read only property)

6.2.36 bitmapFormat as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the bitmap format of the receiver.

Notes: Returns 0 by default. The return value can indicate several different attributes, which are described in Constants.

(Read only property)

6.2.37 bitsPerPixel as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of bits allocated for each pixel in each plane of data.

Notes: This number is normally equal to the number of bits per sample or, if the data is in meshed configuration, the number of bits per sample times the number of samples per pixel. It can be explicitly set to another value (in `initWithBitmapDataPlanes`) in case extra memory is allocated for each pixel. This may be the case, for example, if pixel data is aligned on byte boundaries.

(Read only property)

6.2.38 bytesPerPlane as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of bytes in each plane or channel of data.

Notes: This number is calculated from the number of bytes per row and the height of the image.

(Read only property)

6.2.39 bytesPerRow as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the minimum number of bytes required to specify a scan line (a single row of pixels spanning the width of the image) in each data plane.

Notes: If not explicitly set to another value (in `initWithBitmapDataPlanes`), this number will be figured from the width of the image, the number of bits per sample, and, if the data is in a meshed configuration, the number of samples per pixel. It can be set to another value to indicate that each row of data is aligned on word or other boundaries.

(Read only property)

6.2.40 `CGImage` as Variant

Plugin Version: 13.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the `CGImageMBS` for this image representation.

Notes: If the image was created using a `CGImage`, you can get it back. Else a new `CGImage` may be created for you.

(Read only property)

6.2.41 `colorSpace` as `NSColorSpaceMBS`

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the image rep's `colorSpace`

Notes: Available in Mac OS X v10.6 and later.

(Read only property)

6.2.42 `isPlanar` as Boolean

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns true if image data is a planar configuration and false if its in a meshed configuration.

Notes: In a planar configuration, the image data is segregated into a separate plane for each color and coverage component. In a meshed configuration, the data is integrated into a single plane.

(Read only property)

6.2.43 `numberOfPlanes` as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of separate planes image data is organized into.

Notes: This number is the number of samples per pixel if the data has a separate plane for each component (`isPlanar` returns true) and 1 if the data is meshed (`isPlanar` returns false).

(Read only property)

6.2.44 samplesPerPixel as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of components in the data.

Notes: The returned value includes both color components and the coverage component, if present.

(Read only property)

6.2.45 valueForKey(key as string) as Variant

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The value for the specified property.

Notes: Value can be nil.

(Read and Write computed property)

6.2.46 Constants

Bitmap Format Constants

Constant	Value	Description
NSAlphaFirstBitmapFormat	1	If 0, alpha values are the last component. For example, CMYKA and RGBA. Available in Mac OS X v10.4 and later.
NSAlphaNonpremultipliedBitmapFormat	2	If 0, alpha values are premultiplied. Available in Mac OS X v10.4 and later.
NSFloatingPointSamplesBitmapFormat	4	If 0, samples are integer values. Available in Mac OS X v10.4 and later.

Image format file type constants

Constant	Value	Description
NSBMPFileType	1	Windows bitmap image (BMP) format
NSGIFFileType	2	Graphics Image Format (GIF), originally created by CompuServe for online downloads
NSJPEG2000FileType	5	JPEG 2000 file format.
NSJPEGFileType	3	JPEG format.
NSPNGFileType	4	Portable Network Graphics (PNG) format.
NSTIFFFileType	0	Tagged Image File Format (TIFF)

Image Loading State Constants

Constant	Value	Description
<code>NSImageRepLoadStatusCompleted</code>	-6	Enough data has been provided to successfully decompress the image (less of the complete: flag).
<code>NSImageRepLoadStatusInvalidData</code>	-4	An error occurred during image decompression. The image contains portions of the data that have already been successfully decompressed, if any.
<code>NSImageRepLoadStatusReadingHeader</code>	-2	The image format is known, but not enough data has been read to determine the size, depth, etc., of the image. You should continue to provide more data.
<code>NSImageRepLoadStatusUnexpectedEOF</code>	-5	<code>incrementalLoadFromData</code> was called with <code>true</code> , but not enough data was available for decompression. The image contains the portions of the data that have already been successfully decompressed, if any.
<code>NSImageRepLoadStatusUnknownType</code>	-1	Not enough data to determine image format. You should continue to provide more data.
<code>NSImageRepLoadStatusWillNeedAllData</code>	-3	Incremental loading cannot be supported. Until you call <code>incrementalLoadFromData</code> with <code>true</code> , this status will be returned. You can continue to call the method but no decompression will take place. Once you do call the method with <code>true</code> , then the image will be decompressed and one of the final three messages will be returned.

Tiff Compression Constants

Constant	Value	Description
<code>NSTIFFCompressionCCITTFAX3</code>	3	CCITT Fax Group 3 compression. Used for 1-bit fax images sent over telephone lines.
<code>NSTIFFCompressionCCITTFAX4</code>	4	CCITT Fax Group 4 compression. Used for 1-bit fax images sent over ISDN lines.
<code>NSTIFFCompressionJPEG</code>	6	JPEG compression. No longer supported for input or output.
<code>NSTIFFCompressionLZW</code>	5	LZW compression.
<code>NSTIFFCompressionNEXT</code>	32766	NeXT compressed. Supported for input only.
<code>NSTIFFCompressionNone</code>	1	No compression.
<code>NSTIFFCompressionOldJPEG</code>	32865	Old JPEG compression. No longer supported for input or output.
<code>NSTIFFCompressionPackBits</code>	32773	PackBits compression.

6.3 class NSColorListMBS

6.3.1 class NSColorListMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a Cocoa ColorList.

6.3.2 Methods

6.3.3 colorWithKey(key as string) as NSColorMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color with the given key.

Notes: Returns nil if key doesn't exist.

6.3.4 Create(name as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a color list; specify "" if you don't want a name.

Notes: Note that this does not add the color list to availableColorLists until the color list is saved into the user's path with writeToFile("").

See also:

- 6.3.5 Create(name as string, path as string) as boolean 587

6.3.5 Create(name as string, path as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a color list; specify "" if you don't want a name.

Notes: Note that this does not add the color list to availableColorLists until the color list is saved into the user's path with writeToFile("").

See also:

- 6.3.4 Create(name as string) as boolean 587

6.3.6 insertColor(theColor as NSColorMBS, key as string, index as Integer)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Inserts color at the specified location.

Notes: If a color by the same key is already in the list but at a different location it is removed from there.

6.3.7 isEditable as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether this color list is editable.

Notes: True if it is editable.

6.3.8 Load(name as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Loads the named color list from availableColorLists.

Notes: Returns true on success.

6.3.9 name as string

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Name of the color list.

6.3.10 removeColorWithKey(key as string)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the color with the given key.

Notes: Does nothing if the key does not exist.

6.3.11 removeFile

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: If the color list is in the user's path, removes the corresponding file in user's colorlists directory.

Notes: Also removes the color list from availableColorLists. If there are no outstanding references to the color list this might deallocate the object as well.

6.3.12 setColor(theColor as NSColorMBS, key as string)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: If key already exists, sets the corresponding color; otherwise inserts the color at the end.

6.3.13 writeToFile(path as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes the color list to the given path.

Notes: Use "" to save to the user's private colorlists directory. If the color list is named, this method will also insert the color list into availableColorLists.

Returns true for success.

6.4 class NSColorMBS

6.4.1 class NSColorMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The class for a Cocoa color.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
MsgBox str(c.blueComponent)
```

Blog Entries

- [More Menubar options in macOS Sonoma](#)
- [Styled Text for Labels in your Xojo iOS app](#)
- [MBS Xojo Plugins, version 21.2pr2](#)
- [Basic Forms in MBS Xojo SceneKit Plugin](#)
- [Physics example for SceneKit in Xojo](#)
- [MBS Xojo Plugins, version 19.4pr5](#)
- [Alternating row colors for Dark Mode](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.4](#)
- [Plugin merge/split](#)
- [MBS REALbasic plug-ins version 9.4](#)

Videos

- [Presentation from Xojo Developer Conference 2019 in Miami.](#)

Xojo Developer Magazine

- 19.6, page 50: [Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes](#)
- 19.6, pages 46 to 48: [Rotating Christmas Tree, Using the MBS Plugins to create an animated 3D Christmas tree by Stefanie Juchmes](#)
- 18.4, pages 39 to 41: [On the Scene Again \(Part 3\), Getting Started with SceneKit by Stefanie Juchmes](#)
- 18.1, pages 35 to 37: [On the Scene, Getting Started with SceneKit by Stefanie Juchmes](#)

- 18.1, pages 32 to 33: [On the Scene, Getting Started with SceneKit by Stefanie Juchmes](#)
- 18.1, pages 26 to 27: [On the Scene, Getting Started with SceneKit by Stefanie Juchmes](#)
- 17.5, page 40: [What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes](#)
- 17.1, page 10: [News](#)
- 15.5, pages 39 to 40: [Touch@, Using Apple's Touch Bar with Xojo by Marc Zeedar](#)
- 15.5, page 37: [Touch@, Using Apple's Touch Bar with Xojo by Marc Zeedar](#)

6.4.2 Methods

6.4.3 `alternateSelectedControlColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the face of a selected control.

Example:

```
dim c as NSColorMBS = NSColorMBS.alternateSelectedControlColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for the face of a selected control—a control being clicked or dragged. This color can be used where iApp-like highlighting is desired.

6.4.4 `alternateSelectedControlTextColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text in a selected control.

Example:

```
dim c as NSColorMBS = NSColorMBS.alternateSelectedControlTextColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for text in a selected control—a control being clicked or dragged. This color can be used where iApp-like highlighting is desired.

6.4.5 `alternatingContentBackgroundColors` as `NSColorMBS()`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background colors for alternating content items: such as table view rows, collection view items.

Notes: Available on MacOS 10.14 or newer.

This function calls `controlAlternatingRowBackgroundColors` when `alternatingContentBackgroundColors` is not available.

6.4.6 `blackColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an `NSColor` object whose grayscale value is 0.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.blackColor
```

```
window1.Title=c.description
window1.HasBackColor=true
window1.backcolor=c.colorValue
```

6.4.7 `blendedColorWithFraction(alpha as Double, c as NSColorMBS)` as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an `NSColor` object whose component values are a weighted sum of the receiver's and the specified color object's.

Example:

```
dim b as NSColorMBS = NSColorMBS.blueColor
dim r as NSColorMBS = NSColorMBS.redColor
dim c as NSColorMBS = b.blendedColorWithFraction(0.5, r)
```

```
MsgBox c.description
```

Notes: fraction: The amount of the color to blend with the receiver's color. The method converts color and a copy of the receiver to RGB, and then sets each component of the returned color to fraction of color's value plus 1 - fraction of the receiver's.

color: The color to blend with the receiver's color.

The resulting color object or nil if the colors can't be converted.

6.4.8 blueColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 0.0, 0.0, 1.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.9 brownColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 0.6, 0.4, 0.2 and whose alpha value is 1.0.

Notes: dim c as NSColorMBS = NSColorMBS.brownColor

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.10 CGColorHandle as Integer

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Converts NSColor to CGColor.

Notes: Return value may be an approximation in some cases, so there isn't guaranteed round-trip fidelity. Available on Mac OS X 10.8.

6.4.11 clearColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose grayscale and alpha values are both 0.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.clearColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.12 colorFromPasteboard as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the NSColor currently on the given pasteboard.

Notes: The color currently on the pasteboard or nil if pasteBoard doesn't contain color data.

6.4.13 colorNamed(colorName as String) as NSColorMBS

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

Function: Create catalog colors from values stored with the given name in the Asset Catalog of the main app bundle.

6.4.14 colorSpace as NSColorSpaceMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an object representing the color space of the receiver.

Example:

```
dim b as NSColorMBS = NSColorMBS.blueColor
```

```
MsgBox b.colorSpace.localizedName
```

Notes: An object representing a color space. The returned NSColorSpace object may represent a custom color space.

Calling this method raises an exception if the receiver is not based on a color space represented by an `NSColorSpace` object—specifically, colors designated by `NSNamedColorSpace` and `NSPatternColorSpace`. If you are unsure about a color object, convert it to an equivalent `NSColorSpace`-based object before calling this method. Color objects created with color-space names `NSCalibratedWhiteColorSpace`, `NSCalibratedBlackColorSpace`, `NSCalibratedRGBColorSpace`, `NSDeviceWhiteColorSpace`, `NSDeviceBlackColorSpace`, `NSDeviceRGBColorSpace`, `NSDeviceCMYKColorSpace`, or `NSCustomColorSpace`—or with the `NSColorSpace` class methods corresponding to these names—are safe to use with this method. See “About Color Spaces” in *Color Programming Topics for Cocoa* for a list of these corresponding methods.

6.4.15 `colorUsingColorSpace(colorSpace as NSColorSpaceMBS) as NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a new color object representing the color of the receiver in the specified color space.

Example:

```
dim b as NSColorMBS = NSColorMBS.blueColor
dim c as NSColorMBS = b.colorUsingColorSpace(NSColorSpaceMBS.deviceRGBColorSpace)
```

MsgBox b.description

Notes: space: The color space of the new `NSColor` object.

Returns the new `NSColor` object. This method converts the receiver’s color to an equivalent one in the new color space. Although the new color might have different component values, it looks the same as the original. Returns nil if conversion is not possible.

If the receiver’s color space is the same as that specified in space, this method returns the same `NSColor` object.

6.4.16 `colorUsingColorSpaceName(colorSpace as string) as NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an `NSColor` whose color is the same as the receiver’s, except that the new `NSColor` is in the specified color space.

Notes: colorSpace: The name of the color space containing the new `NSColor` object. If colorSpace is "", the most appropriate color space is used.

The new `NSColor` object or nil if the specified conversion cannot be done.

6.4.17 colorUsingType(type as Integer) as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Convert the color to a color of different type.

Notes: Return nil if conversion is not possible.

6.4.18 colorWithAlphaComponent(alpha as Double) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an NSColor object that has the same color space and component values as the receiver, but the specified alpha component.

Example:

```
dim b as NSColorMBS = NSColorMBS.blueColor.colorWithAlphaComponent(0.5)
MsgBox b.description // shows "NSCalibratedRGBColorSpace 0 0 1 0.5"
```

Notes: alpha: The opacity value of the new NSColor object.

Returns a new NSColor object. If the receiver's color space doesn't include an alpha component, the receiver is returned.

A subclass with explicit opacity components should override this method to return a color with the specified alpha.

6.4.19 colorWithCalibratedHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSColor object using the given opacity and HSB color space components.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithCalibratedHSV(1.0, 0.5, 0.2, 1.0)
MsgBox b.description // shows "NSCalibratedRGBColorSpace 0.2 0.1 0.1 1"
```

Notes: hue: The hue component of the color object in the HSB color space.

saturation: The saturation component of the color object in the HSB color space.

brightness: The brightness (or value) component of the color object in the HSB color space.

alpha: The opacity value of the color object,

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

6.4.20 `colorWithCalibratedRGB`(red as Double, green as Double, blue as Double, alpha as Double=1.0) as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an `NSColor` object using the given opacity and RGB components.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithCalibratedRGB(1.0, 0.5, 0.2, 1.0)
MsgBox b.description // shows "NSCalibratedRGBColorSpace 1 0.5 0.2 1"
```

Notes: red: The red component of the color object.

green: The green component of the color object.

blue: The blue component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

6.4.21 `colorWithCalibratedWhite`(white as Double, alpha as Double=1.0) as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an `NSColor` object using the given opacity and grayscale value.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithCalibratedWhite(1.0, 1.0)
MsgBox b.description // shows "NSCalibratedWhiteColorSpace 1 1"
```

Notes: white: The grayscale value of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

6.4.22 `colorWithCatalogName(listName as String, colorName as String) as NSColorMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Look up and return catalog colors from standard color catalogs, or from `NSColorList` whose name matches the specified catalog name.

6.4.23 `colorWithCGColor(CGColorHandle as Integer) as NSColorMBS`

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: Converts `CGColor` to `NSColor`.

Notes: Return value may be an approximation in some cases, so there isn't guaranteed round-trip fidelity. Available on Mac OS X 10.8.

6.4.24 `colorWithColorSpace(ColorSpace as NSColorSpaceMBS, components() as Double) as NSColorMBS`

Plugin Version: 15.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an `NSColor` object created from the specified components of the given color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericRGBColorSpace
MsgBox n.localizedName
```

```
dim co as NSColorMBS = NSColorMBS.colorWithColorSpace(n, 1.0, 0, 0, 1.0)
MsgBox co.description
```

Notes: `space`: An `NSColorSpace` object representing a color space. The method raises an exception if this is `nil`.

`components`: An array of the components in the specified color space to use to create the `NSColor` object. The order of these components is determined by the color-space profile, with the alpha component always last. (If you want the created color to be opaque, specify 1.0 for the alpha component.)

Returns the color object. If `space` represents a color space that cannot be used with `NSColor` objects (for example, a "pattern" color space) the method returns `nil`.

Raises an `NSExceptionMBS` if the number of components does not match.

See also:

- 6.4.25 `colorWithColorSpace(ColorSpace as NSColorSpaceMBS, paramarray components as Double) as NSColorMBS` 599

6.4.25 `colorWithColorSpace(ColorSpace as NSColorSpaceMBS, paramarray components as Double) as NSColorMBS`

Plugin Version: 15.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColor object created from the specified components of the given color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericRGBColorSpace
MsgBox n.localizedName
```

```
dim co as NSColorMBS = NSColorMBS.colorWithColorSpace(n, 1.0, 0, 0, 1.0)
MsgBox co.description
```

Notes: space: An NSColorSpace object representing a color space. The method raises an exception if this is nil.

components: An array of the components in the specified color space to use to create the NSColor object. The order of these components is determined by the color-space profile, with the alpha component always last. (If you want the created color to be opaque, specify 1.0 for the alpha component.)

Returns the color object. If space represents a color space that cannot be used with NSColor objects (for example, a "pattern" color space) the method returns nil.

Raises an NSExceptionMBS if the number of components does not match.

See also:

- 6.4.24 `colorWithColorSpace(ColorSpace as NSColorSpaceMBS, components() as Double) as NSColorMBS` 598

6.4.26 `colorWithColorSpaceHSV(ColorSpace as NSColorSpaceMBS, hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0) as NSColorMBS`

Plugin Version: 16.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSColor object with the specified color space, hue, saturation, brightness, and alpha channel values.

Notes: ColorSpace: An NSColorSpace object representing a color space. An exception is raised if the color model of the provided color space is not RGB.

hue: The hue (color) component, expressed as a floating-point value in the range 0–1.0.

saturation: The color saturation component, expressed as a floating-point value in the range 0–1.0.

brightness: The brightness component, expressed as a floating-point value in the range 0–1.0.

alpha: The alpha (opacity), expressed as a floating-point value in the range 0 (transparent) to 1.0 (opaque).

Returns the color object.

6.4.27 `colorWithDeviceCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double=1.0)` as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an `NSColor` object using the given opacity value and CMYK components.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(0.5, 1.0, 0.5, 0.2, 1.0)
MsgBox b.description // shows "NSDeviceCMYKColorSpace 0.5 1 0.5 0.2 1"
```

Notes: cyan: The cyan component of the color object.

magenta: The magenta component of the color object.

yellow: The yellow component of the color object.

black: The black component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setcmykcolor`.

6.4.28 `colorWithDeviceHSV(hue as Double, saturation as Double, brightness as Double, alpha as Double=1.0)` as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an `NSColor` object using the given opacity value and HSB color space components.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithDeviceHSV(1.0, 0.5, 0.2, 1.0)
MsgBox b.description // shows "NSDeviceRGBColorSpace 0.2 0.1 0.1 1"
```

Notes: hue: The hue component of the color object.

saturation: The saturation component of the color object.

brightness: The brightness component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setrgbcolor`.

6.4.29 colorWithDeviceRGB(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an NSColor object using the given opacity value and RGB components.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithDeviceRGB(1.0, 0.5, 0.2, 1.0)
MsgBox b.description // shows "NSDeviceRGBColorSpace 1 0.5 0.2 1"
```

Notes: red: The red component of the color object.

green: The green component of the color object.

blue: The blue component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator setrgbcolor.

6.4.30 colorWithDeviceWhite(white as Double, alpha as Double=1.0) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an NSColor object using the given opacity and grayscale values.

Example:

```
dim b as NSColorMBS = NSColorMBS.colorWithDeviceWhite(1.0, 1.0)
MsgBox b.description // shows "NSDeviceWhiteColorSpace 1 1"
```

Notes: white: The grayscale value of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator setgray.

6.4.31 colorWithDisplayP3(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS

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

Function: Returns a color created from the specified components in the Display P3 colorspace.

Notes: red: The red component of the color object.

green: The green component of the color object.

blue: The blue component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

6.4.32 `colorWithGenericGamma22White(white as Double, alpha as Double=1.0)` as `NSColorMBS`

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Returns an color created with the specified white and alpha values in the GenericGamma22 colorspace.

Notes: white: The white value of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

Available in OS X v10.7 and later.

6.4.33 `colorWithHSV(hue as double, saturation as double, brightness as double, alpha as double=1.0)` as `NSColorMBS`

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

Function: Create component-based colors that are compatible with sRGB colorspace.

Notes: These methods are effectively same as `colorWithSRGBHSV:`, but provided for easier reuse of code that uses `UIColor` on iOS. If red, green, blue, or saturation, brightness, or white values are outside of the 0..1 range, these will create colors in the extended sRGB (or for `colorWithWhite:alpha:`, extendedGenericGamma22GrayColorSpace) color spaces. This behavior is compatible with iOS `UIColor`.

6.4.34 `colorWithPatternImage(image as Variant)` as `NSColorMBS`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Creates and returns an `NSColor` object that uses the specified image pattern.

Example:

```
dim p as Picture = LogoMBS(100)
dim n as new NSImageMBS(p)
dim c as NSColorMBS = NSColorMBS.colorWithPatternImage(n)
```

```
dim x as NSImageMBS = c.patternImage
```

```
Backdrop=x.CopyPictureWithMask
```

Notes: The image is tiled starting at the bottom of the window. The image is not scaled.

Parameter is a NSImageMBS object. We declare it as a variant to reduce plugin dependencies.

6.4.35 colorWithRGB(red as double, green as double, blue as double, alpha as double=1.0) as NSColorMBS

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

Function: Create component-based colors that are compatible with sRGB colorspace.

Notes: These methods are effectively same as colorWithSRGBHSV:, but provided for easier reuse of code that uses UIColor on iOS. If red, green, blue, or saturation, brightness, or white values are outside of the 0..1 range, these will create colors in the extended sRGB (or for colorWithWhite:alpha:, extendedGenericGamma22GrayColorSpace) color spaces. This behavior is compatible with iOS UIColor.

6.4.36 colorWithSRGB(red as Double, green as Double, blue as Double, alpha as Double=1.0) as NSColorMBS

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

Function: Returns a color created from the specified components in the sRGB colorspace.

Notes: red: The red component of the color object.

green: The green component of the color object.

blue: The blue component of the color object.

alpha: The opacity value of the color object.

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

Available in OS X v10.7 and later.

6.4.37 colorWithSystemEffect(systemEffect as Integer) as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a color representing the base color with a system defined effect applied to it.

Notes: This color is safe to create before draw time, as the resolution of the final color only happens when being set, retrieving its CGColor, resolving with colorWithType, etc. The return color type is named.

Available on MacOS 10.14 or newer.

6.4.38 colorWithWhite(white as double, alpha as double=1.0) as NSColorMBS

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

Function: Create component-based colors that are compatible with sRGB colorspace.

Notes: These methods are effectively same as colorWithSRGBHSV:, but provided for easier reuse of code that uses UIColor on iOS. If red, green, blue, or saturation, brightness, or white values are outside of the 0..1 range, these will create colors in the extended sRGB (or for colorWithWhite:alpha:, extendedGenericGamma22GrayColorSpace) color spaces. This behavior is compatible with iOS UIColor.

6.4.39 Components as Double()

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the components of the receiver as an array.

Example:

```
dim b as NSColorMBS = NSColorMBS.blueColor
dim c(-1) as Double = b.Components
dim lines(-1) as string
```

```
for each d as Double in c
lines.Append str(d)
next
```

```
MsgBox Join(lines) // shows 0 0 1 1
```

Notes: You can invoke this method on NSColor objects created from custom color spaces to get the individual floating point components, including alpha. Raises an exception if the receiver doesn't have floating-point components. To find out how many components are in the components array, send the receiver a numberOfComponents message.

6.4.40 Constructor(c as color)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Initializes the color with generic colorspace.

Example:

```
dim r as NSColorMBS = NSColorMBS.redColor
```

```
dim c as color = &cFF0000
dim n as new NSColorMBS(c)
```

```
Break // check debugger
```

See also:

- 6.4.41 Constructor(red as Double, green as Double, blue as Double, alpha as Double = 1.0) 605

6.4.41 Constructor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Initializes the color with generic colorspace.

Notes: Pass values in range from 0.0 to 1.0.

See also:

- 6.4.40 Constructor(c as color) 604

6.4.42 controlAccentColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A dynamic color that reflects the user's current preferred accent color.

Notes: This color automatically updates when the accent color preference changes. Do not make assumptions about the color space of this color, which may change across releases.

Available on MacOS 10.14 or newer.

6.4.43 controlAlternatingRowBackgroundColors as NSColorMBS()

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background colors for alternating content items: such as table view rows, collection view items.

Notes: Available on MacOS 10.3 or newer.

6.4.44 controlBackgroundColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the background of large controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlBackgroundColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for the background of large controls such as browsers, table views, and clip views.

6.4.45 controlColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the flat surfaces of a control.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for the flat surfaces of a control. By default, the control color is a pattern color that will draw the ruled lines for the window background, which is the same as returned by `window.BackgroundColor`.

If you use `controlColor` assuming that it is a solid, you may have an incorrect appearance. You should use `lightGrayColor` in its place.

6.4.46 controlDarkShadowColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the dark edge of the shadow dropped from controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlDarkShadowColor
```

```
window1.Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: Of the two dark borders that run along the bottom and right of controls, representing shadows, the color of the outer, darker border.

6.4.47 controlHighlightColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the highlighted bezels of controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlHighlightColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: Of the two light borders that run along the top and left of controls, representing reflections from a light source in the upper left, the color of the inner, duller border.

6.4.48 controlLightHighlightColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for light highlights in controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlLightHighlightColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: Of the two light borders that run along the top and left of controls, representing reflections from a light source in the upper left, the color of the outer, brighter border.

6.4.49 controlShadowColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the shadows dropped from controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlShadowColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: Of the two dark borders that run along the bottom and right of controls, representing shadows, the color of the inner, lighter border.

6.4.50 controlTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text on controls that aren't disabled.

Example:

```
dim c as NSColorMBS = NSColorMBS.controlTextColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.51 currentControlTint as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns current system control tint.

6.4.52 cyanColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 0.0, 1.0, 1.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.cyanColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.53 darkGrayColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose grayscale value is 1/3 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.darkGrayColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.54 disabledControlTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text on disabled controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.disabledControlTextColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.55 findHighlightColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Background color of find indicators: the bubbles that show inline search result hits.

Notes: Requires MacOS 10.13 or newer.

6.4.56 `getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's CMYK and opacity values.

Example:

```
dim co as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(1.0, 0.5, 1.0, 0.8)
dim c,m,y,k as Double
```

```
co.getCMYK(c,m,y,k)
```

```
MsgBox str(c)+" "+str(m)+" "+str(y)+" "+str(k)
```

Notes: cyan: Upon return, contains the cyan component of the color object.

magenta: Upon return, contains the magenta component of the color object.

yellow: Upon return, contains the yellow component of the color object.

black: Upon return, contains the black component of the color object.

alpha: Upon return, contains opacity value of the color object. (optionally)

This method works only with objects representing colors in the `NSDeviceCMYKColorSpace`. Sending it to other objects raises an exception.

See also:

- 6.4.57 `getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double, byref alpha as Double)` 610

6.4.57 `getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double, byref alpha as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's CMYK and opacity values.

Example:

```
dim co as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(1.0, 0.5, 1.0, 0.8, 0.4)
dim c,m,y,k,a as Double
```

```
co.getCMYK(c,m,y,k,a)
```

```
MsgBox str(c)+" "+str(m)+" "+str(y)+" "+str(k)+" "+str(a)
```

Notes: cyan: Upon return, contains the cyan component of the color object.

magenta: Upon return, contains the magenta component of the color object.

yellow: Upon return, contains the yellow component of the color object.

black: Upon return, contains the black component of the color object.

alpha: Upon return, contains opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSDeviceCMYKColorSpace. Sending it to other objects raises an exception.

See also:

- 6.4.56 `getCMYK(byref cyan as Double, byref magenta as Double, byref yellow as Double, byref black as Double)` 610

6.4.58 `getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's HSB component and opacity values in the respective arguments.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithCalibratedHSV(1.0, 0.5, 1.0, 0.4)
dim h,s,v,a as Double
```

```
c.getHSV(h,s,v,a)
```

```
MsgBox str(h)+" "+str(s)+" "+str(v)+" "+str(a)
```

Notes: hue: Upon return, contains the hue component of the color object.

saturation: Upon return, contains the saturation component of the color object.

brightness: Upon return, contains the brightness component of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

See also:

- 6.4.59 `getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double, byref alpha as Double)` 611

6.4.59 `getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double, byref alpha as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's HSB component and opacity values in the respective arguments.

Notes: hue: Upon return, contains the hue component of the color object.

saturation: Upon return, contains the saturation component of the color object.

brightness: Upon return, contains the brightness component of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

See also:

- 6.4.58 `getHSV(byref hue as Double, byref saturation as Double, byref brightness as Double)` 611

6.4.60 `getRGB(byref red as Double, byref green as Double, byref blue as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's RGB component and opacity values in the respective arguments.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
```

```
dim r,g,b as Double
```

```
c.getRGB(r,g,b)
```

```
MsgBox str(r)+" "+str(g)+" "+str(b) // shows "0 0 1"
```

Notes: red: Upon return, contains the red component of the color object.

green: Upon return, contains the green component of the color object.

blue: Upon return, contains the blue component of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

Plugin converts color to calibrated RGB if needed.

See also:

- 6.4.61 `getRGB(byref red as Double, byref green as Double, byref blue as Double, byref alpha as Double)` 613

6.4.61 getRGB(byref red as Double, byref green as Double, byref blue as Double, byref alpha as Double)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's RGB component and opacity values in the respective arguments.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
dim r,g,b,a as Double
```

```
c.getRGB(r,g,b,a)
```

```
MsgBox str(r)+" "+str(g)+" "+str(b)+" "+str(a) // shows "0 0 1 1"
```

Notes: red: Upon return, contains the red component of the color object.

green: Upon return, contains the green component of the color object.

blue: Upon return, contains the blue component of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

Plugin converts color to calibrated RGB if needed.

See also:

- 6.4.60 getRGB(byref red as Double, byref green as Double, byref blue as Double)

612

6.4.62 getWhite(byref white as Double)

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's grayscale value and alpha values.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceWhite(1.0, 1.0)
dim w as Double
```

```
c.getWhite(w)
```

```
MsgBox str(w)
```

Notes: white: Upon return, contains the grayscale value of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedWhiteColorSpace, NSCalibratedBlackColorSpace, NSDeviceBlackColorSpace, or NSDeviceWhiteColorSpace color space. Sending it to other objects raises an exception.

See also:

- 6.4.63 `getWhite(byref white as Double, byref alpha as Double)` 614

6.4.63 `getWhite(byref white as Double, byref alpha as Double)`

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the receiver's grayscale value and alpha values.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceWhite(1.0, 1.0)
dim w,a as Double
```

```
c.getWhite(w,a)
```

```
MsgBox str(w)+" " +str(a)
```

Notes: white: Upon return, contains the grayscale value of the color object.

alpha: Upon return, contains the opacity value of the color object. (optionally)

This method works only with objects representing colors in the NSCalibratedWhiteColorSpace, NSCalibratedBlackColorSpace, NSDeviceBlackColorSpace, or NSDeviceWhiteColorSpace color space. Sending it to other objects raises an exception.

See also:

- 6.4.62 `getWhite(byref white as Double)` 613

6.4.64 `grayColor as NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose grayscale value is 0.5 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.grayColor
```

```
window1.Title=c.description
```

```
window1.HasBackColor=true
```

```
window1.backcolor=c.colorValue
```

6.4.65 greenColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 0.0, 1.0, 0.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.greenColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.66 gridColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the optional gridlines in, for example, a table view.

Example:

```
dim c as NSColorMBS = NSColorMBS.gridColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.67 headerColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used as the background color for header cells in table views and outline views.

Example:

```
dim c as NSColorMBS = NSColorMBS.headerColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used as the background for header cells in table and outline views.

6.4.68 headerTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text in header cells in table views and outline views.

Example:

```
dim c as NSColorMBS = NSColorMBS.headerTextColor
```

```
window1.Title=c.description
window1.HasBackColor=true
window1.backcolor=c.colorValue
```

6.4.69 highlightColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color that represents the virtual light source on the screen.

Example:

```
dim c as NSColorMBS = NSColorMBS.highlightColor
```

```
window1.Title=c.description
window1.HasBackColor=true
window1.backcolor=c.colorValue
```

6.4.70 highlightWithLevel(level as Double) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColor object that represents a blend between the receiver and the highlight color returned by highlightColor.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
dim h as NSColorMBS = c.highlightWithLevel(0.5)
MsgBox h.description
```

Notes: level: The amount of the highlight color that is blended with the receiver's color. This should be a number from 0.0 through 1.0. A highlightLevel below 0.0 is interpreted as 0.0; a highlightLevel above 1.0 is interpreted as 1.0.

Returns the new NSColor object. Returns nil if the colors can't be converted.

Invoke this method when you want to brighten the receiving NSColor for use in highlights.

6.4.71 keyboardFocusIndicatorColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color that represents the keyboard focus ring around controls.

Example:

```
dim c as NSColorMBS = NSColorMBS.keyboardFocusIndicatorColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.72 knobColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the flat surface of a slider knob that hasn't been selected.

Example:

```
dim c as NSColorMBS = NSColorMBS.knobColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The knob's beveled edges, which set it in relief, are drawn in highlighted and shadowed versions of the face color. When a knob is selected, its color changes to selectedKnobColor.

6.4.73 labelColor as NSColorMBS

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

Function: Foreground color for static text and related elements.

Notes: Requires MacOS 10.10 or newer.

6.4.74 lightGrayColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose grayscale value is 2/3 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.lightGrayColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.75 linkColor as NSColorMBS

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

Function: Foreground color for standard system links

Notes: Requires MacOS 10.10 or newer.

6.4.76 magentaColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 1.0, 0.0, 1.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.magentaColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.77 orangeColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 1.0, 0.5, 0.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.orangeColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.78 patternImage as Variant

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the image that the receiver is using as a pattern.

Example:

```
dim p as Picture = LogoMBS(100)  
dim n as new NSImageMBS(p)  
dim c as NSColorMBS = NSColorMBS.colorWithPatternImage(n)  
dim x as NSImageMBS = c.patternImage
```

```
Backdrop=x.CopyPictureWithMask
```

Notes: The image used by the color object. If the receiver doesn't have an image, this method raises an exception.

Declared a variant instead of NSImageMBS to reduce plugin dependencies.

6.4.79 placeholderTextColor as NSColorMBS

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

Function: Foreground color for placeholder text in controls or text views.

Notes: Requires MacOS 10.10 or newer.

6.4.80 purpleColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 0.5, 0.0, 0.5 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.purpleColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.81 quaternaryLabelColor as NSColorMBS

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

Function: Foreground color for large secondary or disabled static text, separators, large glyphs/icons, etc.

Notes: Requires MacOS 10.10 or newer.

6.4.82 redColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 1.0, 0.0, 0.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.redColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.83 scrollBarColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for scroll "bars"—that is, for the groove in which a scroller's knob moves.

Example:

```
dim c as NSColorMBS = NSColorMBS.scrollBarColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for scroll bars.

6.4.84 scrubberTexturedBackgroundColor as NSColorMBS

Plugin Version: 16.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The scrubber texture background color.

6.4.85 secondaryLabelColor as NSColorMBS

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

Function: Foreground color for secondary static text and related elements.

Notes: Requires MacOS 10.10 or newer.

6.4.86 secondarySelectedControlColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used in non-key views.

Example:

```
dim c as NSColorMBS = NSColorMBS.secondarySelectedControlColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.87 selectedContentBackgroundColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background color of selected and emphasized (focused) content: table views rows, collection views, etc.

Notes: Alias for `alternateSelectedControlColor`
Requires MacOS 10.14 or newer.

6.4.88 `selectedControlColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the face of a selected control.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedControlColor
```

```
window1.Title=c.description  
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for the face of a selected control—a control being dragged or clicked.

6.4.89 `selectedControlTextColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text in a selected control—a control being clicked or dragged.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedControlTextColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.90 `selectedKnobColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the slider knob when it is selected.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedKnobColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for a slider knob that is selected—that is, dragged.

6.4.91 selectedMenuItemColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the face of selected menu items.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedMenuItemColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for selected menu items.

6.4.92 selectedMenuItemTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the text in menu items.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedMenuItemTextColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for text in selected menu items.

6.4.93 selectedTextBackgroundColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the background of selected text.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedTextBackgroundColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.94 selectedTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for selected text.

Example:

```
dim c as NSColorMBS = NSColorMBS.selectedTextColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.95 separatorColor as NSColorMBS

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

Function: Foreground used for separators between sections of content.

Notes: Not appropriate for use as split view or window chrome dividers.

Requires MacOS 10.14 or newer.

6.4.96 shadowColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color that represents the virtual shadows cast by raised objects on the screen.

Example:

```
dim c as NSColorMBS = NSColorMBS.shadowColor
```

```
Title=c.description
```

```
window1.HasBackColor=true
```

```
window1.backcolor=c.colorValue
```

6.4.97 shadowWithLevel(level as Double) as NSColorMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColor object that represents a blend between the receiver and the shadow color returned by shadowColor.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
```

```
dim h as NSColorMBS = c.shadowWithLevel(0.5)
```

```
MsgBox h.description
```

Notes: level: The amount of the shadow color used for the blend. This should be a number from 0.0 through 1.0. A shadowLevel below 0.0 is interpreted as 0.0; a shadowLevel above 1.0 is interpreted as 1.0.

Returns the new NSColor object. Returns nil if the colors can't be converted.

Invoke this method when you want to darken the receiving NSColor for use in shadows.

6.4.98 systemBlueColor as NSColorMBS

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

Function: The system blue color.

Notes: Requires MacOS 10.10 or newer.

6.4.99 systemBrownColor as NSColorMBS

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The system brown color.

Notes: Requires MacOS 10.10 or newer.

6.4.100 `systemGrayColor` as `NSColorMBS`

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

Function: The system gray color.

Notes: Requires MacOS 10.10 or newer.

6.4.101 `systemGreenColor` as `NSColorMBS`

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

Function: The system green color.

Notes: Requires MacOS 10.10 or newer.

6.4.102 `systemIndigoColor` as `NSColorMBS`

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The system indigo color.

Notes: Available in MacOS 10.15 or later.

6.4.103 systemOrangeColor as NSColorMBS

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

Function: The system orange color.

Notes: Requires MacOS 10.10 or newer.

6.4.104 systemPinkColor as NSColorMBS

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

Function: The system pink color.

Notes: Requires MacOS 10.10 or newer.

6.4.105 systemPurpleColor as NSColorMBS

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

Function: The system purple color.

Notes: Requires MacOS 10.10 or newer.

6.4.106 systemRedColor as NSColorMBS

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

Function: The system red color.

Notes: Requires MacOS 10.10 or newer.

6.4.107 systemTealColor as NSColorMBS

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The system teal color.

Notes: Available in MacOS 10.12 or later.

6.4.108 systemYellowColor as NSColorMBS

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

Function: The system yellow color.

Notes: Requires MacOS 10.10 or newer.

6.4.109 tertiaryLabelColor as NSColorMBS

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

Function: Foreground color for disabled static text and related elements.

Notes: Requires MacOS 10.10 or newer.

6.4.110 textBackgroundColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the text background.

Example:

```
dim c as NSColorMBS = NSColorMBS.textBackgroundColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for the background of text. When text is selected, its background color changes to the return value of selectedTextBackgroundColor.

6.4.111 textColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for text.

Example:

```
dim c as NSColorMBS = NSColorMBS.textColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

Notes: The system color used for text. When text is selected, its color changes to the return value of `selectedTextColor`.

6.4.112 `underPageBackgroundColor` as `NSColorMBS`

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

Function: Background areas revealed behind documents. This should not be used for drawing, and `NSVisualEffectMaterialUnderPageBackground` should be used instead.

Notes: Requires MacOS 10.8 or newer.

6.4.113 `unemphasizedSelectedContentBackgroundColor` as `NSColorMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background color of selected and unemphasized content: table views rows, collection views, etc.

Notes: Requires MacOS 10.14 or newer.

Alias for `secondarySelectedControlColor`

6.4.114 `unemphasizedSelectedTextBackgroundColor` as `NSColorMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background for unemphasized text selection (e.g. when the associated control/view/window is not key)

Notes: Requires MacOS 10.14 or newer.

6.4.115 `unemphasizedSelectedTextColor` as `NSColorMBS`

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The foreground for unemphasized text selection (e.g. when the associated control/view/window is not key)

Notes: Requires MacOS 10.14 or newer.

6.4.116 `whiteColor` as `NSColorMBS`

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose grayscale and alpha values are both 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.whiteColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.117 windowBackgroundColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a pattern color that will draw the ruled lines for the window background.

Example:

```
dim c as NSColorMBS = NSColorMBS.windowBackgroundColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.118 windowFrameColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for window frames, except for their text.

Example:

```
dim c as NSColorMBS = NSColorMBS.windowFrameColor
```

```
Title=c.description
```

```
window1.HasBackColor=true  
window1.backcolor=c.colorValue
```

6.4.119 windowFrameTextColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the system color used for the text in window frames.

Example:

```
dim c as NSColorMBS = NSColorMBS.windowFrameTextColor

// convert to Calibrated RGB ColorSpace
c=c.colorUsingColorSpaceName("NSCalibratedRGBColorSpace")

Title=c.description

window1.HasBackColor=true
window1.backcolor=c.colorValue
```

6.4.120 writeToPasteboard

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Writes the receiver's data to the specified pasteboard.

6.4.121 yellowColor as NSColorMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns an NSColor object whose RGB value is 1.0, 1.0, 0.0 and whose alpha value is 1.0.

Example:

```
dim c as NSColorMBS = NSColorMBS.yellowColor

Title=c.description

window1.HasBackColor=true
window1.backcolor=c.colorValue
```

6.4.122 Properties

6.4.123 alphaComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The alpha value of the color.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
MsgBox str(c.alphaComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.124 blackComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a CMYK color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(0.1, 0.2, 0.3, 0.4)
MsgBox str(c.blackComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.125 blueComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a RGB color.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
MsgBox str(c.blueComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.126 brightnessComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a HSV color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceHSV(0.1, 0.2, 0.3)
MsgBox str(c.brightnessComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.127 catalogNameComponent as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the name of the catalog containing the receiver's name.

Notes: This method raises an exception if the receiver's color space isn't NSNamedColorSpace.
(Read only property)

6.4.128 colorNameComponent as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's name.

Notes: This method raises an exception if the receiver's color space isn't NSNamedColorSpace.
(Read only property)

6.4.129 colorSpaceName as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Get the color space of the color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceHSV(0.1, 0.2, 0.3)
```

```
MsgBox c.colorSpaceName
```

Notes: (Read only property)

6.4.130 colorValue as color

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Returns the color value of this color (ignoring color space).

Notes: The alphaComponent is ignored.

Returns RGB(RedComponent*255, GreenComponent*255, BlueComponent*255).

Plugin converts color to calibrated RGB if needed.

The color is converted to generic RGB Colorspace if needed as that is the one Xojo uses for pictures.

(Read only property)

6.4.131 cyanComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a CMYK color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(0.1, 0.2, 0.3, 0.4)
MsgBox str(c.cyanComponent)
```

Notes: Value is from 0.0 to 1.0.

(Read only property)

6.4.132 description as string

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: The description of the color object.

Example:

```
dim n as NSColorMBS = NSColorMBS.blueColor
MsgBox n.description // shows "NSCalibratedRGBColorSpace 0 0 1 1"
```

Notes: (Read only property)

6.4.133 greenComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a RGB color.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor  
MsgBox str(c.greenComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.134 Handle as Integer

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: Useful for declares. Value is a NSColor pointer.
(Read and Write property)

6.4.135 hueComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a HSV color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceHSV(0.1, 0.2, 0.3)  
MsgBox str(c.hueComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.136 localizedCatalogNameComponent as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the name of the catalog containing the receiver's name as a localized string.

Notes: This string may be displayed in user interface items like color pickers.
(Read only property)

6.4.137 localizedColorNameComponent as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the name of the receiver as a localized string.

Notes: The name of color object as a localized string. This string may be displayed in user interface items like color pickers.

(Read only property)

6.4.138 magentaComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a CMYK color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(0.1, 0.2, 0.3, 0.4)
MsgBox str(c.magentaComponent)
```

Notes: Value is from 0.0 to 1.0.

(Read only property)

6.4.139 numberOfComponents as Integer

Plugin Version: 9.8, Platform: macOS, Targets: All.

Function: Returns the number of components in the receiver.

Example:

```
dim n as NSColorMBS = NSColorMBS.blueColor
```

```
MsgBox str(n.numberOfComponents) // shows 4
```

Notes: The number of components in the color object. The floating-point components counted include alpha. This method raises an exception if the receiver doesn't have floating-point components.

Available in Mac OS X v10.4 and later.

(Read only property)

6.4.140 redComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a RGB color.

Example:

```
dim c as NSColorMBS = NSColorMBS.blueColor
MsgBox str(c.redComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.141 saturationComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a HSV color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceHSV(0.1, 0.2, 0.3)
MsgBox str(c.saturationComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.142 Type as Integer

Plugin Version: 18.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Return the type of the color, which in turn determines which accessors are valid for this color.

Notes: It's an exception-raising error to use properties that are not valid for a color; so, before using a given accessor on a color, call `colorUsingType` to convert the color to the appropriate type. Or, if wanting to use colorSpace-specific accessors such as `redComponent`, convert to the desired colorSpace with `colorUsingColorSpace`.

Note that as of macOS 10.13, the new APIs `type` and `colorUsingType` replace `colorSpaceName` and `colorUsingColorSpaceName` as the funnel APIs for determining the type of color. In code destined to run on 10.13 and newer systems, subclasses are encouraged to implement these two rather than the older `colorSpaceName` based methods.

Requires MacOS 10.13 or newer.
(Read only property)

6.4.143 whiteComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a Gray color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceWhite(0.1)
MsgBox str(c.whiteComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.144 yellowComponent as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The color value from a CMYK color.

Example:

```
dim c as NSColorMBS = NSColorMBS.colorWithDeviceCMYK(0.1, 0.2, 0.3, 0.4)
MsgBox str(c.yellowComponent)
```

Notes: Value is from 0.0 to 1.0.
(Read only property)

6.4.145 Constants

System Effect

Constant	Value	Description
SystemEffectDeepPressed	2	Deep Pressed
SystemEffectDisabled	3	Disabled
SystemEffectNone	0	No
SystemEffectPressed	1	Pressed
SystemEffectRollover	4	Rollover

Color Types

Constant	Value	Description
TypeCatalog	2	nColor with a catalog name and a color name. These colors can be dynamic and have multiple underlying color values.one
TypeComponentBased	0	Colors which have a color space (specified as NSColorSpace) and appropriate number of CGFloat components.
TypePattern	1	Colors that draw a pattern, specified as an NSImage.

6.5 class NSColorSamplerMBS

6.5.1 class NSColorSamplerMBS

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class for color sampler.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.4](#)
- [MBS Xojo Plugins, version 19.4pr2](#)

Xojo Developer Magazine

- [17.6, page 11: News](#)

6.5.2 Methods

6.5.3 Available as Boolean

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether NSColorSampler is available.

Notes: Returns true on MacOS 10.15 or newer.

6.5.4 Constructor

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

Notes: May raise PlatformNotSupportedException if not available.

6.5.5 Show

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Shows the color sampler.

Notes: Begins or attaches to an existing color sampling session which presents UI to the user for selecting a color from their screen. The handler will be called on the main thread when the user completes the session (either by selection, or cancelation). In the event of user-cancellation, the color passed with be nil.

Calls Completed event later.

6.5.6 Properties

6.5.7 Handle as Integer

Plugin Version: 19.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

6.5.8 Events

6.5.9 Completed(selectedColor as NSColorMBS)

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

Function: The event called when color sampler finished.

Notes: selectedColor is selected color or nil in case of cancel.

6.6 class NSColorSpaceMBS

6.6.1 class NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSColorSpace class enables the creation of objects representing custom color spaces.

Notes: You can make custom color spaces from ColorSync profiles or from ICC profiles. NSColorSpace also has factory methods that return objects representing the system color spaces.

Blog Entries

- [MBS Xojo Plugins, version 24.0pr6](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1](#)
- [MBS Xojo Plugins, version 19.1pr2](#)
- [MBS Xojo Plugins, version 18.5pr3](#)
- [Colorspaces in MacOS with Xojo](#)
- [MBS Xojo / Real Studio plug-ins in version 13.4](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr3](#)
- [Plugin merge/split](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.6](#)

6.6.2 Methods

6.6.3 adobeRGB1998ColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing an Adobe RGB (1998) color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.adobeRGB1998ColorSpace
MsgBox n.localizedName
```

Notes: The NSColorSpace object. This color-additive color space has red, green, blue, and alpha components.

The Adobe RGB (1998) color space was designed to encompass most of the colors achievable on CMYK color printers, but by using RGB primary colors on a device such as the computer display. For more information on this color space, go to this website:

<http://www.adobe.com/digitalimag/adobergb.html>

6.6.4 availableColorSpacesWithModel(Model as Integer) as NSColorSpaceMBS()

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the list of color spaces available on the system that are displayed in the color panel, in the order they are displayed in the color panel.

Example:

```
dim a(-1) as NSColorSpaceMBS
dim m as Integer = NSColorSpaceMBS.NSRGBColorSpaceModel
```

```
a = NSColorSpaceMBS.availableColorSpacesWithModel(m)
```

```
dim names(-1) As string
```

```
for each c as NSColorSpaceMBS in a
names.Append c.localizedName
next
```

```
MsgBox Join(names,EndOfLine)
```

Notes: This method doesn't return color spaces created on the fly or spaces without user-displayable names. Pass NSUnknownColorSpaceModel as model to get all available color spaces.

Available in Mac OS X v10.6 and later.

6.6.5 CGColorSpaceHandle as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Core Graphics color-space object that represents a color space equivalent to the receiver's.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox hex(n.CGColorSpaceHandle)
```

Notes: A reference to an Core Graphics color-space object (CGColorSpaceRef) or 0 if the type of color space represented by the receiver cannot be represented by a CGColorSpace object.

Available in Mac OS X version 10.5.

6.6.6 `colorSpaceForColorSpaceName(name as string)` as `NSColorSpaceMBS`

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: The colorspace for this color space name.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.colorSpaceForColorSpaceName(NSColorSpaceMBS.NS-  
DeviceWhiteColorSpace)  
MsgBox n.localizedName
```

Notes: This plugin function uses an undocumented method from Apple's AppKit framework. Seems like they use it only internally.

Works well on Mac OS X 10.6.

6.6.7 `ColorSpaceWithCGColorSpace(CGColorSpaceHandle as Integer)` as `NSColorSpaceMBS`

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

Function: Initializes and returns an `NSColorSpace` object initialized from a Core Graphics color-space object.

Notes: The initialized `NSColorSpace` object or nil if initialization was not successful, which might happen if the color space represented by the `CGColorSpace` object is not supported by `NSColorSpace`.

Because `NSColorSpace` might retain or copy the `CGColorSpace` object depending on circumstances, you should not assume pointer equality of the provided object with that returned by `CGColorSpace`. And even if the pointer equality is preserved during runtime, it may not be after the `NSColorSpace` object is archived and unarchived.

Available in Mac OS X version 10.5 and later.

6.6.8 `ColorSpaceWithColorSyncProfile(ColorSyncProfileHandle as Integer)` as `NSColorSpaceMBS`

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

Function: Initializes and returns an NSColorSpace object given a ColorSync profile.

Notes: The initialized NSColorSpace object or nil if initialization was not successful.

Available in Mac OS X v10.4 and later.

6.6.9 ColorSpaceWithICCProfileData(File as FolderItem) as NSColorSpaceMBS

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

Function: Initializes and returns an NSColorSpace object given an ICC profile.

Notes: file: The path to the ICC profile to use when initializing the NSColorSpace object. For information on ICC profiles, see the latest ICC specification at the International Color Consortium website.

Returns the initialized NSColorSpace object or nil if initialization was not successful.

Available in Mac OS X v10.4 and later.

See also:

- 6.6.10 ColorSpaceWithICCProfileData(ICCProfileData as Memoryblock) as NSColorSpaceMBS 645

6.6.10 ColorSpaceWithICCProfileData(ICCProfileData as Memoryblock) as NSColorSpaceMBS

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

Function: Initializes and returns an NSColorSpace object given an ICC profile.

Notes: ICCProfileData:

The ICC profile to use when initializing the NSColorSpace object. For information on ICC profiles, see the latest ICC specification at the International Color Consortium website.

Returns the initialized NSColorSpace object or nil if initialization was not successful.

Available in Mac OS X v10.4 and later.

See also:

- 6.6.9 ColorSpaceWithICCProfileData(File as FolderItem) as NSColorSpaceMBS 645

6.6.11 colorSyncProfileHandle as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the ColorSync profile from which the receiver was created.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox hex(n.colorSyncProfileHandle)
```

Notes: The ColorSync profile on which the receiver is based. You need to cast this value to an object of opaque type `CMProfileRef`. Returns `NULL` if the receiver was created from a ICC-profile data instead. See ColorSync Manager Reference for further information on `CMProfileRef`.

Available in Mac OS X v10.4 and later.

6.6.12 Constructor(`ICCProfileData` as `Memoryblock`)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes and returns an `NSColorSpace` object given an ICC profile.

Notes: `ICCProfileData`:

The ICC profile to use when initializing the `NSColorSpace` object. For information on ICC profiles, see the latest ICC specification at the International Color Consortium website.

The initialized `NSColorSpace` object or `nil` if initialization was not successful.

Available in Mac OS X v10.4 and later.

6.6.13 `deviceCMYKColorSpace` as `NSColorSpaceMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an `NSColorSpace` object representing a calibrated or device-dependent CMYK color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.deviceCMYKColorSpace
MsgBox n.localizedName
```

Notes: The `NSColorSpace` object. This color space has cyan, magenta, yellow, black, and alpha components. Typical devices that use the color-subtractive CMYK color space are color printers. This object corresponds to the Cocoa color space name `NSDeviceCMYKColorSpace`.

Available in Mac OS X v10.4 and later.

6.6.14 deviceGrayColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a calibrated or device-dependent gray color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.deviceGrayColorSpace
MsgBox n.localizedName
```

Notes: The NSColorSpace object. The color space also includes an alpha component. Typical devices that use this color space are grayscale printers and displays. This object corresponds to the Cocoa color space name NSDeviceWhiteColorSpace.

Available in Mac OS X v10.4 and later.

6.6.15 deviceRGBColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a calibrated or device-dependent RGB color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.deviceRGBColorSpace
MsgBox n.localizedName
```

Notes: The NSColorSpace object. This color space has red, green, blue, and alpha components. Typical devices that use the color-additive RGB color space are displays and scanners. This object corresponds to the Cocoa color space name NSDeviceRGBColorSpace.

6.6.16 genericCMYKColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a device-independent CMYK color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericCMYKColorSpace
MsgBox n.localizedName
```

Notes: The NSColorSpace object. This color space has cyan, magenta, yellow, black and alpha component.

Available in Mac OS X v10.4 and later.

6.6.17 genericGamma22GrayColorSpace as NSColorSpaceMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a gray color space with a gamma value of 2.2.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericGamma22GrayColorSpace  
MsgBox n.localizedName
```

Notes: Available in Mac OS X v10.6 and later.

6.6.18 genericGrayColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a device-independent gray color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericGrayColorSpace  
MsgBox n.localizedName
```

Notes: The NSColorSpace object. The color space also includes an alpha component. This object corresponds to the Cocoa color space name NSCalibratedWhiteColorSpace.

Available in Mac OS X v10.4 and later.

6.6.19 genericRGBColorSpace as NSColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSColorSpace object representing a device-independent RGB color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericRGBColorSpace  
MsgBox n.localizedName
```

Notes: The NSColorSpace object. This color-additive color space has red, green, blue, and alpha components. This object corresponds to the Cocoa color space name NSCalibratedRGBColorSpace.

Available in Mac OS X v10.4 and later.

6.6.20 ICCProfileData as Memoryblock

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the ICC profile data from which the receiver was created.

Notes: The ICC profile from which the receiver was created. This method attempts to compute the profile data from a CMProfileRef object and returns nil if it is unable to.

For information on ICC profiles, see the latest ICC specification at the International Color Consortium website.

Available in Mac OS X v10.4 and later.

6.6.21 initWithCGColorSpace(CGColorSpaceHandle as Integer)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes and returns an NSColorSpace object initialized from a Core Graphics color-space object.

Notes: The initialized NSColorSpace object or nil if initialization was not successful, which might happen if the color space represented by the CGColorSpace object is not supported by NSColorSpace.

Because NSColorSpace might retain or copy the CGColorSpace object depending on circumstances, you should not assume pointer equality of the provided object with that returned by CGColorSpace. And even if the pointer equality is preserved during runtime, it may not be after the NSColorSpace object is archived and unarchived.

Available in Mac OS X version 10.5 and later.

6.6.22 initWithColorSyncProfile(ColorSyncProfileHandle as Integer)

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes and returns an `NSColorSpace` object given a `ColorSync` profile.

Notes: The initialized `NSColorSpace` object or `nil` if initialization was not successful.

Available in Mac OS X v10.4 and later.

6.6.23 `sRGBColorSpace` as `NSColorSpaceMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an `NSColorSpace` object representing an sRGB color space.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox n.localizedName
```

Notes: The `NSColorSpace` object. This color-additive color space has red, green, blue, and alpha components.

The sRGB color space is a standard color space for use on monitors, printers, and the Internet. For further information on sRGB, see this website:

<http://www.color.org/srgb.html>

Available in Mac OS X version 10.5.

6.6.24 Properties

6.6.25 `colorSpaceModel` as `Integer`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the model on which the color space of the receiver is based.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox str(n.colorSpaceModel)
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

6.6.26 colorSpaceName as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: The name of the colorspace.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.genericRGBColorSpace  
MsgBox n.colorSpaceName
```

Notes: This plugin function uses an undocumented method from Apple's AppKit framework. Seems like they use it only internally.

Works well on Mac OS X 10.6.

Returns an empty string for some color spaces.

(Read only property)

6.6.27 description as string

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the description of the color space.

Notes: (Read only property)

6.6.28 File as FolderItem

Plugin Version: 18.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to file if available.

Notes: Will be nil for profiles created from string/MemoryBlock.

(Read only property)

6.6.29 Handle as Integer

Plugin Version: 17.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

6.6.30 `localizedName` as string

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the localized name of the receiver.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox n.localizedName
```

Notes: The name of the color space as a localized string or nil if no localized name exists.

Available in Mac OS X v10.4 and later.
(Read only property)

6.6.31 `numberOfColorComponents` as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of components supported by the receiver.

Example:

```
dim n as NSColorSpaceMBS = NSColorSpaceMBS.sRGBColorSpace
MsgBox str(n.numberOfColorComponents) // shows 3
```

Notes: The number of components (excluding alpha) the receiver supports or zero if the receiver is not based on float components.

Available in Mac OS X v10.4 and later.
(Read only property)

6.6.32 Constants

Constants

Constant	Value	Description
NSCalibratedBlackColorSpace	"NSCalibratedBlackColorSpace"	Calibrated color space with black and alpha components. Deprecated in Mac OS X v10.6.
NSCalibratedRGBColorSpace	"NSCalibratedRGBColorSpace"	Calibrated color space with red, green, blue, and alpha components. You can also create a color with HSB (hue, saturation, and brightness) components and can extract these components.
NSCalibratedWhiteColorSpace	"NSCalibratedWhiteColorSpace"	Calibrated color space with white and alpha components.
NSCMYKColorSpaceModel	2	The CYMK (cyan, yellow, magenta, black) color-space model. Can refer to both device-dependent and generic color spaces. Available in Mac OS X v10.4 and later.
NSCustomColorSpace	"NSCustomColorSpace"	Custom NSColorSpace object and floating-point components in that space. A custom color-space object represents a color space not predefined by the Application Kit. See "Working With Color" for information on creating custom color-space objects.
NSDeviceBlackColorSpace	"NSDeviceBlackColorSpace"	Device-dependent color space with black and alpha components. Available in Mac OS X v10.4 and later.
NSDeviceCMYKColorSpace	"NSDeviceCMYKColorSpace"	Device-dependent color space with cyan, magenta, yellow, and black components.
NSDeviceNColorSpaceModel	4	DeviceN is a color-space model from Adobe Systems used in PostScript and PDF color specification. Available in Mac OS X v10.4 and later.
NSDeviceRGBColorSpace	"NSDeviceRGBColorSpace"	Device-dependent color space with red, green, blue, and alpha components. You can also create a color with HSB (hue, saturation, and brightness) components and can extract these components.
NSDeviceWhiteColorSpace	"NSDeviceWhiteColorSpace"	Device-dependent color space with white and alpha components. Available in Mac OS X v10.4 and later.
NSGrayColorSpaceModel	0	The grayscale color-space model. Can refer to both device-dependent and generic color space variants. Available in Mac OS X v10.4 and later.
NSIndexedColorSpaceModel	5	An indexed color space, which identifies specified colors by index number. An indexed color value (a color specification in indexed color) is an index value that refers to a color in a color list. Available in Mac OS X version 10.5 and later.
NSLABColorSpaceModel	3	The L*a*b* device-independent color-space model, which is relative to a reference white point. Available in Mac OS X v10.4 and later.
NSNamedColorSpace	"NSNamedColorSpace"	Catalog name and color name components. The components of this color space are indexes into lists of named colors. The catalogs of named colors come with lookup tables to generate the correct color on a given device.
NSPatternColorSpace	"NSPatternColorSpace"	Pattern image (tiled) identifies a pattern color space, which is simply an image that is repeated and over again in a tiled pattern.
NSPatternColorSpaceModel	6	Identifies a pattern color space, which is simply an image that is repeated and over again in a tiled pattern. Available in Mac OS X version 10.5 and later.
NSRGBColorSpaceModel	1	The RGB (red green blue) color-space model. Can refer to both device-dependent and generic color spaces. Available in Mac OS X v10.4 and later.
NSUnknownColorSpaceModel	-1	This model is not known to NSColorSpace. Available in Mac OS X v10.4 and later.

6.7 class NSEPSImageRepMBS

6.7.1 class NSEPSImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** An NSEPSImageRep object can render an image from encapsulated PostScript (EPS) code.

Notes: Won't work in macOS Sonoma since Apple removed the feature from the OS.

Subclass of the NSImageRepMBS class.

Blog Entries

- [MBS Xojo Plugins, version 23.3pr2](#)
- [EPS support deprecated in macOS Sonoma](#)
- [MBS Xojo Plugins, version 18.1pr2](#)
- [MBS Real Studio Plugins, version 12.0pr5](#)

6.7.2 Methods

6.7.3 boundingBox as NSRectMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the rectangle that bounds the receiver.

Notes: This rectangle is obtained from the "%%BoundingBox:" comment in the EPS header when the NSEPSImageRep object is initialized.

6.7.4 Constructor(data as Memoryblock)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes an NSEPSImageRep object initialized with the specified EPS data.

Notes: The size of the receiver is set using the bounding box information specified in the EPS header comments.

On success the handle property is not zero.

6.7.5 EPSRepresentation as Memoryblock

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the EPS representation of the receiver.

6.7.6 imageRepWithData(data as Memoryblock) as NSEPSImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSEPSImageRep object initialized with the specified EPS data.

Notes: Returns a new, initialized NSEPSImageRep object or nil if the object could not be initialized.

The size of the receiver is set using the bounding box information specified in the EPS header comments.

6.7.7 prepareGState

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Implemented by subclasses to configure the graphics state prior to drawing.

Notes: The draw method of NSEPSImageRep sends this message to itself just before rendering the EPS code. The default implementation of this method does nothing. You can override it in your subclass to prepare the graphics state as needed.

6.7.8 Properties**6.7.9 pdfImage as NSPDFImageRepMBS**

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The PDF representation of the image.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("logo_new.eps")
```

```
// read in as image
```

```
dim image as new nsimageMBS(f)
```

```
// get representations
```

```
dim reps() as NSImageRepMBS = image.representations
```

```
// get EPS representation
```

```
dim rep as NSEPSImageRepMBS = NSEPSImageRepMBS(Reps(0))
```

```
// get PDF representation
```

```
dim pdf as NSPDFImageRepMBS = rep.pdfImage
```

Break

Notes: When you load an EPS file, a PDF file is automatically created.

6.8 class NSImageMBS

6.8.1 class NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: An NSImage object is a high-level class for manipulating image data.

Notes: You use this class to load existing images or create new ones and composite them into a view or other image. This class works in conjunction with one or more image representation objects (subclasses of NSImageRep), which manage the actual image data.

All methods in this class will catch exceptions from Cocoa and raise a NSExcptionMBS instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

Blog Entries

- [MBS Xojo Plugins in version 22.4](#)
- [News from the MBS Xojo Plugins Version 22.1](#)
- [Several ways for picture to PDF in MBS Plugins](#)
- [News from the MBS Xojo Plugins Version 20.5](#)
- [IconFamilyMBS class deprecated](#)
- [Custom menu checkmarks](#)
- [NSImage and image orientation](#)
- [Tip of the day: Rotate image with CoreImage](#)
- [MBS REALbasic Plugins Version 10.4 release notes](#)
- [MBS REALbasic plug-in 9.6](#)

Videos

- [Apple MapView In Xojo](#)

Xojo Developer Magazine

- 6.4, page 34: [Creating PDF Files, How to create PDFs using the MBS Plugins](#) by Christian Schmitz
- 5.6, pages 32 to 33: [Third Party Plugins: Statusitems, Getting an icon in the top right of the Mac OS X menubar](#) by Christian Schmitz
- 20.3, page 80: [Great Shots With Continuity Camera, Use your iOS device to take a picture for your Mac](#) by Stefanie Juchmes

- 20.3, page 10: News
- 19.1, page 91: Maps Part 8, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter
- 19.1, page 82: Maps Part 8, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter
- 19.1, page 80: Maps Part 8, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter
- 19.1, page 78: Maps Part 8, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter
- 18.6, page 73: MapKit Part 7, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter
- 18.6, pages 68 to 69: MapKit Part 7, Implementing Maps in Xojo desktop apps with the MapKitMBS plugin by Markus Winter

6.8.2 Methods

6.8.3 addRepresentation(img as NSImageRepMBS)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds the specified image representation object to to the receiver.

Notes: After invoking this method, you may need to explicitly set features of the new image representation, such as the size, number of colors, and so on. This is true particularly when the NSImage object has multiple image representations to choose from. See NSImageRep and its subclasses for the methods you use to complete initialization.

Any representation added by this method is retained by the receiver. Image representations cannot be shared among multiple NSImage objects.

6.8.4 BMPRepresentation as Memoryblock

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The image as the binary data in a BMP file.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream
```

```
p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p,p.Mask)

f=SpecialFolder.Desktop.Child("test.bmp")
b=f.CreateBinaryFile("")
b.Write img.BMPRepresentation
b.Close

f.Launch
```

Notes: BMP does not support masks.

6.8.5 BMPRepresentationMT as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The image as the binary data in a BMP file.

Notes: BMP does not support masks.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.6 cancelIncrementalLoad

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Cancels the current download operation immediately, if the image is being incrementally loaded.

Notes: This call has no effect if the image is not loading.

6.8.7 canInitWithPasteboard as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Tests whether the receiver can create an instance of itself using pasteboard data.

Notes: This method uses the NSImageRep class method `imageUnfilteredPasteboardTypes` to find a class

that can handle the data in the specified pasteboard. If you create your own `NSImageRep` subclasses, override the `imageUnfilteredPasteboardTypes` method to notify `NSImage` of the pasteboard types your class supports.

6.8.8 Constructor

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The default constructor creating a dummy `NSImage` object.

See also:

- 6.8.9 `Constructor(data as Memoryblock)` 660
- 6.8.10 `Constructor(file as folderitem)` 660
- 6.8.11 `Constructor(image as Picture, mask as picture = nil)` 661
- 6.8.12 `Constructor(width as Double, height as Double)` 661

6.8.9 Constructor(data as Memoryblock)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The data constructor calling `initWithData` internally.

Notes: Calls `initWithData`.

On success the image is valid and the handle is not zero.

See also:

- 6.8.8 `Constructor` 660
- 6.8.10 `Constructor(file as folderitem)` 660
- 6.8.11 `Constructor(image as Picture, mask as picture = nil)` 661
- 6.8.12 `Constructor(width as Double, height as Double)` 661

6.8.10 Constructor(file as folderitem)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The file constructor calling `initWithContentsOfURL(file)` internally.

Notes: Calls `initWithContentsOfURL`.

On success the image is valid and the handle is not zero.

See also:

- 6.8.8 `Constructor` 660
- 6.8.9 `Constructor(data as Memoryblock)` 660

6.8. CLASS NSIMAGEMBS	661
• 6.8.11 Constructor(image as Picture, mask as picture = nil)	661
• 6.8.12 Constructor(width as Double, height as Double)	661

6.8.11 Constructor(image as Picture, mask as picture = nil)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates a new NSImageMBS object based on the image data in a Xojo picture.

Example:

```
dim img as new NSImageMBS(pict)
```

Notes: Optional you can pass a picture with the mask. It is valid to use the mask property of the image for the second parameter.

With 11.3 plugins we are deprecating to pass a mask. The plugin prefers to simply take the mask or alpha channel of the picture itself.

On success the image is valid and the handle is not zero.

Calls initWithPicture.

See also:

• 6.8.8 Constructor	660
• 6.8.9 Constructor(data as Memoryblock)	660
• 6.8.10 Constructor(file as folderitem)	660
• 6.8.12 Constructor(width as Double, height as Double)	661

6.8.12 Constructor(width as Double, height as Double)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The size constructor calling initWithSize internally.

Notes: Calls initWithSize.

On success the image is valid and the handle is not zero.

See also:

• 6.8.8 Constructor	660
• 6.8.9 Constructor(data as Memoryblock)	660
• 6.8.10 Constructor(file as folderitem)	660
• 6.8.11 Constructor(image as Picture, mask as picture = nil)	661

6.8.13 CopyMask as picture

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Copies the content of the NSImage in current size.

Example:

```
dim f as FolderItem
dim n as NSImageMBS

f=SpecialFolder.Desktop.Child("test.png")
n=new NSImageMBS(f)
Backdrop=n.CopyMask
```

Notes: Copies only the alpha channel as mask.
Returns nil on any error.

Not implemented for iOS.

6.8.14 CopyPicture(CGColorSpace as Variant = nil, BackgroundColor as NSColorMBS = nil) as picture

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Copies the content of the NSImage in current size.

Example:

```
dim f as FolderItem
dim n as NSImageMBS

f=SpecialFolder.Desktop.Child("test.png")
n=new NSImageMBS(f)
Backdrop=n.CopyPicture // background is black
```

Notes: A convenience function instead of using CGPictureContextMBS with DrawIntoCGContextAtRect.

Returns nil on any error.

With Colorspace parameter you can pass a RGB CGColorSpace to define which colorspace is used. Default is DeviceRGB, but you could also pass generic RGB or other.

If BackgroundColor is not nil, the image is filled in background with this color and NSImage rendered on

top of it.

Implemented for iOS in v22.4 to return underlying UIImage as Xojo picture. Colorspace and background are ignored as picture is passed directly.

6.8.15 CopyPictureRect(x as Integer, y as Integer, w as Integer, h as Integer, CGColorSpace as Variant = nil, BackgroundColor as NSColorMBS = nil) as picture

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

Function: Copies the content of the NSImage from the given rectangle.

Example:

```
dim f as FolderItem
dim n as NSImageMBS

f=SpecialFolder.Desktop.Child("test.png")
n=new NSImageMBS(f)
Backdrop=n.CopyPicture // background is black
```

Notes: A convenience function instead of using CGPictureContextMBS with DrawIntoCGContextAtRect.

Returns nil on any error.

With Colorspace parameter you can pass a RGB CGColorSpace to define which colorspace is used. Default is DeviceRGB, but you could also pass generic RGB or other.

If BackgroundColor is not nil, the image is filled in background with this color and NSImage rendered on top of it.

Not implemented for iOS.

6.8.16 CopyPictureWithAlpha as picture

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Copies the content of the NSImage in current size.

Notes: This is a function for Cocoa target which returns picture with alpha channel.

Returns nil on any error.

Not implemented for iOS.

6.8.17 CopyPictureWithAlphaRect(x as Integer, y as Integer, w as Integer, h as Integer) as picture

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Copies the content of the NSImage in given rectangle.

Notes: This is a function for Cocoa target which returns picture with alpha channel.

Returns nil on any error.

Not implemented for iOS.

6.8.18 CopyPictureWithMask(CGColorSpace as Variant = nil) as picture

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Copies the content of the NSImage in current size.

Example:

```
dim f as FolderItem
```

```
dim n as NSImageMBS
```

```
f=SpecialFolder.Desktop.Child("test.png")
```

```
n=new NSImageMBS(f)
```

```
Backdrop=n.CopyPictureWithMask
```

Notes: Copies the picture and its mask.

This function is faster than CopyPicture and CopyMask combined as the picture is only copied one time to an internal buffer.

A convenience function instead of using CGContextMBS with DrawIntoCGContextAtRect.

Returns nil on any error.

With Colorspace parameter you can pass a RGB CGColorSpace to define which colorspace is used. Default is DeviceRGB, but you could also pass generic RGB or other.

Not implemented for iOS.

6.8.19 DrawIntoCGContextAtPoint(cgcontext as Integer, x as Double, y as Double, sx as Double, sy as Double, SourceW as Double, SourceH as Double, operation as Integer, fraction as Double) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Draws the image.

Notes: Same as DrawIntoCGContextAtRect, but with a point instead of a rectangle.

6.8.20 DrawIntoCGContextAtRect(cgcontext as Integer, x as Double, y as Double, w as Double, h as Double, SourceX as Double, SourceY as Double, SourceW as Double, SourceH as Double, operation as Integer, fraction as Double) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Draws the image.

Example:

```
Function OpenAsNSImage(extends file as folderitem) As picture
dim width as Integer
dim height as Integer
dim c as NSImageMBS
dim g as CGPictureContextMBS

// load image from file
c=new NSImageMBS

// is that image valid?
if c.initWithContentsOfURL(file) then
Width=c.Width
height=c.Height

// create a drawing buffer to draw inside
g=new CGPictureContextMBS(width,height)

if g.Handle<>0 then // valid?

// now draw the image inside.
// you could scale or even apply transparency...
if c.DrawIntoCGContextAtRect(g.Handle, 0, 0, width, height, 0,0,width,height,2,1.0) then
// make a RB Picture from it
Return g.CopyPicture
```

```

end if
end if
end if
End Function

```

Notes: Draws the image into a `CGContext`. You need to specify first the destination rectangle followed by the source rectangle.

fraction:

The opacity of the image, specified as a value from 0.0 to 1.0. Specifying a value of 0.0 draws the image as fully transparent while a value of 1.0 draws the image as fully opaque. Values greater than 1.0 are interpreted as 1.0.

operation codes:

<code>NSCompositeClear</code>	= 0	Transparent.
<code>NSCompositeCopy</code>	= 1	Source image.
<code>NSCompositeSourceOver</code>	= 2	Source image wherever source image is opaque, and destination image elsewhere.
<code>NSCompositeSourceIn</code>	= 3	Source image wherever both images are opaque, and transparent elsewhere.
<code>NSCompositeSourceOut</code>	= 4	Source image wherever source image is opaque but destination image is transparent, and transparent elsewhere.
<code>NSCompositeSourceAtop</code>	= 5	Source image wherever both images are opaque, destination image wherever destination image is opaque but source image is transparent, and transparent elsewhere.
<code>NSCompositeDestinationOver</code>	= 6	Destination image wherever destination image is opaque, and source image elsewhere.
<code>NSCompositeDestinationIn</code>	= 7	Destination image wherever both images are opaque, and transparent elsewhere.
<code>NSCompositeDestinationOut</code>	= 8	Destination image wherever destination image is opaque but source image is transparent, and transparent elsewhere.
<code>NSCompositeDestinationAtop</code>	= 9	Destination image wherever both images are opaque, source image wherever source image is opaque but destination image is transparent, and transparent elsewhere.
<code>NSCompositeXOR</code>	= 10	Exclusive OR of source and destination images.
<code>NSCompositePlusDarker</code>	= 11	Sum of source and destination images, with color values approaching 0 as a limit.
<code>NSCompositeHighlight</code>	= 12	Source image wherever source image is opaque, and destination image elsewhere.
<code>NSCompositePlusLighter</code>	= 13	Sum of source and destination images, with color values approaching 1 as a limit.

Returns true on success and false on failure.

6.8.21 GIFRepresentation as Memoryblock

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The image as the binary data in a GIF file.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p,p.Mask)

f=SpecialFolder.Desktop.Child("test.gif")
b=f.CreateBinaryFile("")
b.Write img.GIFRepresentation
b.Close

f.Launch
```

Notes: GIF does support masks in a limited way.

6.8.22 GIFRepresentationMT as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The image as the binary data in a GIF file.

Notes: GIF does support masks in a limited way.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.23 imageByFadingToFraction(fraction as Double) as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns an image with a mask faded to the given percentage

Example:

```

dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

img=img.imageByFadingToFraction(0.1)

f=SpecialFolder.Desktop.Child("test.png")
b=f.CreateBinaryFile("")
b.Write img.PNGRepresentation
b.Close

f.Launch

```

Notes: Returns nil on failure.

6.8.24 imageByScalingToSize(width as Double, height as Double) as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Scales image to the new size.

Example:

```

dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

img=img.imageByScalingToSize(200,200)

```

```
f=SpecialFolder.Desktop.Child("test.png")
b=f.CreateBinaryFile("")
b.Write img.PNGRepresentation
b.Close

f.Launch
```

Notes: Returns nil on failure.
Internally creates a copy of the image.
See also:

- 6.8.25 imageByScalingToSize(width as Double, height as Double, fraction as Double) as NSImageMBS
669
- 6.8.26 imageByScalingToSize(width as Double, height as Double, fraction as Double, flip as boolean, proportionally as boolean) as NSImageMBS
670

6.8.25 imageByScalingToSize(width as Double, height as Double, fraction as Double) as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Scales image to the new size with given fading.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

img=img.imageByScalingToSize(200,200,0.5)

f=SpecialFolder.Desktop.Child("test.png")
b=f.CreateBinaryFile("")
b.Write img.PNGRepresentation
b.Close

f.Launch
```

Notes: Returns nil on failure.
Internally creates a copy of the image.
See also:

- 6.8.24 `imageByScalingToSize(width as Double, height as Double)` as `NSImageMBS` 668
- 6.8.26 `imageByScalingToSize(width as Double, height as Double, fraction as Double, flip as boolean, proportionally as boolean)` as `NSImageMBS` 670

6.8.26 `imageByScalingToSize(width as Double, height as Double, fraction as Double, flip as boolean, proportionally as boolean)` as `NSImageMBS`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Scales image to the new size with given fading fraction.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

img=img.imageByScalingToSize(200,200)

f=SpecialFolder.Desktop.Child("test.png")
b=f.CreateBinaryFile("")
b.Write img.PNGRepresentation
b.Close

f.Launch
```

Notes: Returns nil on failure.
The image can be flipped vertically with the flip property.
Internally creates a copy of the image.
See also:

- 6.8.24 `imageByScalingToSize(width as Double, height as Double)` as `NSImageMBS` 668

- 6.8.25 imageByScalingToSize(width as Double, height as Double, fraction as Double) as NSImageMBS
669

6.8.27 imageFileTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of strings identifying the image types supported by the registered NSImageRep objects.

Example:

```
dim types() as string = NSImageMBS.imageFileTypes  
MsgBox Join(types, EndOfLine)
```

Notes: An array of strings, each of which identifies a single supported file type. The array can include encoded HFS file types as well as filename extensions.

This list includes all file types supported by registered subclasses of NSImageRep plus those that can be converted to a supported type by a user-installed filter service. You can pass the array returned by this method directly to NSOpenPanelMBS.

When creating a subclass of NSImageRep, do not override this method. Instead, override the imageUnfilteredFileTypes method to notify NSImage of the file types your class supports directly.

6.8.28 imageNamed(name as string) as NSImageMBS

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns the NSImage instance associated with the specified name.

Notes: This method searches for named images in several places, returning the first image it finds matching the given name. The order of the search is as follows:

1. Search for an object whose name was set explicitly using the setName: method and currently resides in the image cache.
2. Search the application's main bundle for a file whose name matches the specified string. (For information on how the bundle is searched, see "Searching for Bundle Resources" in Bundle Programming Guide.)
3. Search the Application Kit framework for a shared image with the specified name.

When looking for files in the application bundle, it is better (but not required) to include the filename extension in the name parameter. When naming an image with the setName method, it is also convention not to include filename extensions in the names you specify. That way, you can easily distinguish between images you have named explicitly and those you want to load from the application's bundle.

One particularly useful image you can retrieve is your application's icon. This image is set by Cocoa automatically and referenced by the string "NSApplicationIcon". Icons for other applications can be obtained through the use of methods declared in the NSWorkspace class. You can also retrieve some standard system images using Cocoa defined constants; for more information, see the Constants section of this class.

If an application is linked in Mac OS X v10.5 or later, images requested using this method and whose name ends in the word "Template" are automatically marked as template images.

The NSImage class keeps a reference to any named images in a table until the image name is cleared. Consequently you do not need to retain the returned image object unless its name could be cleared. You can clear an image object from the table by passing nil to the setName: method of the corresponding NSImage object.

Here is a good list of identifiers you can use:
http://hetima.github.io/fucking_nsimage_syntax/

6.8.29 imagePasteboardTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of strings identifying the pasteboard types supported directly by the registered NSImageRep objects.

Example:

```
dim types() as string = NSImageMBS.imagePasteboardTypes
MsgBox Join(types, EndOfLine)
```

Notes: Returns an array of strings, each of which identifies a single supported pasteboard type. By default, this list contains the NSPDFPboardType, NSPICTPboardType, NSPostScriptPboardType, and NSTIFFPboardType types.

This list includes all pasteboard types supported by registered subclasses of NSImageRep plus those that can be converted to a supported type by a user-installed filter service.

When creating a subclass of NSImageRep, do not override this method. Instead, override the imageUnfilteredPasteboardTypes method to notify NSImage of the pasteboard types your class supports.

6.8.30 imageTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of UTI strings identifying the image types supported by the registered NSImageRep objects, either directly or through a user-installed filter service.

Example:

```
dim types() as string = NSImageMBS.imageTypes
MsgBox Join(types, EndOfLine)
```

Notes: Returns an array of strings, each of which contains a UTI identifying a supported image type. Some sample image-related UTI strings include "public.image", "public.jpeg", and "public.tiff". For a list of supported types, see `UTCORETYPES.H`.

The returned list includes UTIs all file types supported by registered subclasses of NSImageRep plus those that can be converted to a supported type by a user-installed filter service. You can use the returned UTI strings with any method that supports UTIs.

You should not override this method directly. Instead, you should override the `imageTypes` method of NSImageRep.

Available in Mac OS X v10.5 and later.

6.8.31 imageUnfilteredFileTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of strings identifying the file types supported directly by the registered NSImageRep objects.

Example:

```
dim types() as string = NSImageMBS.imageUnfilteredFileTypes
MsgBox Join(types, EndOfLine)
```

Notes: An array of strings, each of which identifies a single supported file type. File types are identified by file extension and HFS file types.

The returned list does not contain pasteboard types that are available only through a user-installed filter service.

6.8.32 imageUnfilteredPasteboardTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of strings identifying the pasteboard types supported directly by the registered NSImageRep objects.

Example:

```
dim types() as string = NSImageMBS.imageUnfilteredPasteboardTypes
MsgBox Join(types, EndOfLine)
```

Notes: An array of strings, each of which identifies a single supported pasteboard type. The returned list does not contain pasteboard types that are supported only through a user-installed filter service.

6.8.33 imageUnfilteredTypes as string()

Plugin Version: 12.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of UTI strings identifying the image types supported directly by the registered NSImageRep objects.

Example:

```
dim types() as string = NSImageMBS.imageUnfilteredTypes
MsgBox Join(types, EndOfLine)
```

Notes: Returns an array of strings, each of which contains a UTI identifying a supported image type. Some sample image-related UTI strings include "public.image", "public.jpeg", and "public.tiff". For a list of supported types, see `UTCoreTypes.h`.

The returned list includes UTI strings only for those file types that are supported directly by registered subclasses of NSImageRep. It does not include types that are supported through user-installed filter services. You can use the returned UTI strings with any method that supports UTIs.

You should not override this method directly. Instead, you should override the `imageUnfilteredTypes` method of NSImageRep.

6.8.34 imageWithCGImage(CGImage as Variant, width as Double = 0, height as Double = 0) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the CGImage.

Example:

```
dim logo as Picture = LogoMBS(500)
dim cgimage as CGImageMBS = CGCreateImageMBS(logo)
dim nsimage as NSImageMBS = NSImageMBS.imageWithCGImage(cgimage)
dim pic as Picture = nsimage.CopyPictureWithMask
Backdrop = pic
```

Notes: If width is zero, we take the width from the CGImage.
If height is zero, we take the height from the CGImage.

cgImage: The source CGImageMBS object.
width & height: The size of the new image.

Returns an initialized NSImage instance, or nil if the new instance cannot be initialized.

You should not assume anything about the image, other than that drawing it is equivalent to drawing the CGImage.
Available in OS X v10.6 and later.

6.8.35 imageWithContentsOfFile(file as folderitem) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified file.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("mbs.jpg")
dim image as NSImageMBS = NSImageMBS.imageWithContentsOfFile(file)
Backdrop = image.CopyPictureWithMask
```

Notes: File: The file to open.

Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified file.

6.8.36 imageWithContentsOfFileMT(file as folderitem) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified file.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("mbs.jpg")
dim image as NSImageMBS = NSImageMBS.imageWithContentsOfFileMT(file)
Backdrop = image.CopyPictureWithMask
```

Notes: File: The file to open.

Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified file.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.37 imageWithContentsOfPath(path as string) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified file.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("mbs.jpg")
dim path as string = file.NativePath
dim image as NSImageMBS = NSImageMBS.imageWithContentsOfPath(path)
Backdrop = image.CopyPictureWithMask
```

Notes: path: A full or relative path name specifying the file with the desired image data. Relative paths must be relative to the current working directory.

Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified file.

The filename parameter should include the file extension that identifies the type of the image data. This method looks for an NSImageRep subclass that handles that data type from among those registered with

NSImage.

6.8.38 imageWithContentsOfPathMT(path as string) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified file.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("mbs.jpg")
dim path as string = file.NativePath
dim image as NSImageMBS = NSImageMBS.imageWithContentsOfPathMT(path)
Backdrop = image.CopyPictureWithMask
```

Notes: path: A full or relative path name specifying the file with the desired image data. Relative paths must be relative to the current working directory.

Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified file.

The filename parameter should include the file extension that identifies the type of the image data. This method looks for an NSImageRep subclass that handles that data type from among those registered with NSImage.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.39 imageWithContentsOfURL(URL as string) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified URL.

Example:

```
dim url as string = "http://www.monkeybreadsoftware.de/images/MBSLogo.jpg"
dim img as NSImageMBS = NSImageMBS.imageWithContentsOfURL(url)
Backdrop = img.CopyPictureWithMask
```

Notes: Returns an initialized `NSImage` instance, or `nil` if the method cannot create an image representation from the contents of the specified URL.

6.8.40 `imageWithContentsOfURLMT(URL as string)` as `NSImageMBS`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSImage` instance with the contents of the specified URL.

Example:

```
dim url as string = "http://www.monkeybreadsoftware.de/images/MBSLogo.jpg"
dim img as NSImageMBS = NSImageMBS.imageWithContentsOfURLMT(url)
Backdrop = img.CopyPictureWithMask
```

Notes: Returns an initialized `NSImage` instance, or `nil` if the method cannot create an image representation from the contents of the specified URL.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

6.8.41 `imageWithData(data as memoryblock)` as `NSImageMBS`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSImage` instance with the contents of the specified memoryblock.

Example:

```
dim logo as Picture = LogoMBS(500)
dim jpeg as string = PictureToJPEGStringMBS(logo, 75)
dim nsimage as NSImageMBS = NSImageMBS.imageWithData(jpeg)
dim pic as Picture = nsimage.CopyPictureWithMask
Backdrop = pic
```

Notes: Returns an initialized `NSImage` instance, or `nil` if the method cannot create an image representation from the contents of the specified data object.

See also:

- 6.8.42 `imageWithData(data as string)` as `NSImageMBS`

6.8.42 imageWithData(data as string) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified string.

Example:

```
dim logo as Picture = LogoMBS(500)
dim jpeg as string = PictureToJPEGStringMBS(logo, 75)
dim nsimage as NSImageMBS = NSImageMBS.imageWithData(jpeg)
dim pic as Picture = nsimage.CopyPictureWithMask
Backdrop = pic
```

Notes: Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified data object.

See also:

- 6.8.41 imageWithData(data as memoryblock) as NSImageMBS

678

6.8.43 imageWithDataMT(data as memoryblock) as NSImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified memoryblock.

Example:

```
dim logo as Picture = LogoMBS(500)
dim jpeg as string = PictureToJPEGStringMBS(logo, 75)
dim nsimage as NSImageMBS = NSImageMBS.imageWithDataMT(jpeg)
dim pic as Picture = nsimage.CopyPictureWithMask
Backdrop = pic
```

Notes: Returns an initialized NSImage instance, or nil if the method cannot create an image representation from the contents of the specified data object.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

See also:

- 6.8.44 imageWithDataMT(data as string) as NSImageMBS

680

6.8.44 `imageWithDataMT(data as string)` as `NSImageMBS`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSImage` instance with the contents of the specified string.

Example:

```
dim logo as Picture = LogoMBS(500)
dim jpeg as string = PictureToJPEGStringMBS(logo, 75)
dim nsimage as NSImageMBS = NSImageMBS.imageWithDataMT(jpeg)
dim pic as Picture = nsimage.CopyPictureWithMask
Backdrop = pic
```

Notes: Returns an initialized `NSImage` instance, or `nil` if the method cannot create an image representation from the contents of the specified data object.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

See also:

- 6.8.43 `imageWithDataMT(data as memoryblock)` as `NSImageMBS`

679

6.8.45 `imageWithHandle(Handle as Integer)` as `NSImageMBS`

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

Function: Creates a new picture for a `NSImage` handle.

6.8.46 `imageWithSymbolConfiguration(configuration as NSImageSymbolConfigurationMBS)` as `NSImageMBS`

Plugin Version: 22.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new symbol image with the specified configuration.

Example:

```
Dim config As NSImageSymbolConfigurationMBS = NSImageSymbolConfigurationMBS.configurationWith-
  PointSize(40, 0)
Dim n As NSImageMBS = NSImageMBS.imageWithSystemSymbolName("trash")
Dim o As NSImageMBS = n.imageWithSymbolConfiguration(config)
```

```
Backdrop = o.CopyPictureWithAlpha
```

6.8.47 `imageWithSystemSymbolName(name as string, accessibilityDescription as string = "") as NSImageMBS`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Returns an image for a given symbol name.

Example:

```
Dim n As NSImageMBS = NSImageMBS.imageWithSystemSymbolName("trash")
If n <> Nil Then
```

```
n.setSize(300,300)
Backdrop = n.CopyPictureWithAlpha
```

```
End If
```

Notes: For macOS 11.0 or newer.

See list of symbols in SF Symbol font, see SF Symbols app or the list here:
<https://developer.apple.com/design/human-interface-guidelines/sf-symbols/overview/>

6.8.48 `imageWithTintColor(tintColor as NSColorMBS) as NSImageMBS`

Plugin Version: 20.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a copy of the image colored with the given tint.

Notes: Input image should be grayscale.

6.8.49 `initWithContentsOfURL(file as folderitem) as boolean`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes and returns an NSImage instance with the contents of the specified folderitem.

Notes: Returns true on success and false on failure.

See also:

- 6.8.50 `initWithContentsOfURL(url as string) as boolean`

6.8.50 `initWithContentsOfURL(url as string)` as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes and returns an `NSImage` instance with the contents of the specified URL.

Notes: Returns true on success and false on failure.

See also:

- 6.8.49 `initWithContentsOfURL(file as folderitem)` as boolean

681

6.8.51 `initWithData(data as Memoryblock)` as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes and with the contents of the specified data in the string.

Notes: Returns true on success and false on failure.

6.8.52 `initWithDataIgnoringOrientation(data as Memoryblock)` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes and returns an `NSImage` instance with the contents of the specified memoryblock, ignoring the EXIF orientation tags.

Notes: An initialized `NSImage` instance, or nil if the method cannot create an image representation from the contents of the specified data object.

Available in Mac OS X v10.6 and later.

6.8.53 `initWithIconRef(IconHandle as Integer)` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes the image object with a Carbon-style icon resource.

Notes: `IconHandle`: A reference to a Carbon icon resource (`IconRef`).

Creates one or more bitmap image representations, one for each size icon contained in the `IconRef` data structure. This initialization method automatically retains the data in the `iconRef` parameter and loads the bitmaps from that data file lazily.

Available in Mac OS X v10.5 and later.

6.8.54 `initWithPasteboard` as boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Initializes an NSImage instance with data from the pasteboard.

Notes: The specified pasteboard should contain a type supported by one of the registered NSImageRep subclasses. Table 1 lists the default pasteboard types and file extensions for several NSImageRep subclasses.

6.8.55 initWithPicture(img as picture, mask as picture = nil) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes the image object with image data from the Xojo pictures.

Example:

```
dim img as new NSImageMBS
if img.initWithPicture(pict, pict.mask) then
  MsgBox "OK"
end if
```

Notes: Optional you can pass a picture with the mask. It is valid to use the mask property of the image for the second parameter.

With 11.3 plugins we are deprecating to pass a mask. The plugin prefers to simply take the mask or alpha channel of the picture itself.

On success the image is valid and the handle is not zero.

6.8.56 initWithSize(width as Double, height as Double) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes an empty image object with the given size.

Notes: Returns true on success and false on failure.

6.8.57 JPEGRepresentation as Memoryblock

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The image as the binary data in a JPEG file.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
```

```

dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

f=SpecialFolder.Desktop.Child("test.jpeg")
b=f.CreateBinaryFile("")
b.Write img.JPEGRepresentation
b.Close

f.Launch

```

Notes: JPEG does not support masks. Uses 80% for the quality.

6.8.58 JPEGRepresentationMT as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The image as the binary data in a JPEG file.

Example:

```

// take some Picture
dim logo as Picture = LogoMBS(500)

// make a NSImageMBS from it
dim nsimage as new NSImageMBS(logo)

// use thread friendly compress function
dim jpeg as MemoryBlock = nsimage.JPEGRepresentationMT

// decode to see if it worked
dim test as Picture = JPEGStringToPictureMBS(jpeg)

// and display
Backdrop = test

```

Notes: JPEG does not support masks. Uses 80% for the quality.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will

block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.59 JPEGRepresentationWithCompressionFactor(factor as Double) as Memoryblock

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The image as the binary data in a JPEG file.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p)

f=SpecialFolder.Desktop.Child("test.jpeg")
b=f.CreateBinaryFile("")
b.Write img.JPEGRepresentationWithCompressionFactor(0.01)
b.Close

f.Launch
```

Notes: Factor for compression goes from 0.0 to 1.0.
JPEG does not support masks.

6.8.60 JPEGRepresentationWithCompressionFactorMT(factor as Double) as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The image as the binary data in a JPEG file.

Example:

```
// take some Picture
dim logo as Picture = LogoMBS(500)
```

```
// make a NSImageMBS from it
dim nsimage as new NSImageMBS(logo)

// use thread friendly compress function
dim jpeg as MemoryBlock = nsimage.JPEGRepresentationWithCompressionFactorMT(1.0)

// decode to see if it worked
dim test as Picture = JPEGStringToPictureMBS(jpeg)

// and display
Backdrop = test
```

Notes: Factor for compression goes from 0.0 to 1.0.
JPEG does not support masks.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.61 NSImageHintUserInterfaceLayoutDirection as string

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

Function: One of the dictionary keys used in the hints dictionary.

Notes: value is a number with NSUserInterfaceLayoutDirection enum value

6.8.62 NSImageNameActionTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the imageNamed method.

An action menu template image.

Available in Mac OS X v10.5 and later.

6.8.63 NSImageNameAddTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Example:

```
Dim n As NSImageMBS = NSImageMBS.imageNamed(NSImageMBS.NSImageNameAddTemplate)
```

```
canvas1.backdrop = n.CopyPictureWithAlpha
```

Notes: An add item template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the imageNamed method.

6.8.64 NSImageNameAdvanced as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Example:

```
Dim n As NSImageMBS = NSImageMBS.imageNamed(NSImageMBS.NSImageNameAdvanced)
```

```
canvas1.backdrop = n.CopyPictureWithAlpha
```

Notes: To access this image, pass the specified constant to the imageNamed method.

Advanced preferences toolbar icon. Use in a preferences window only.

Available in Mac OS X v10.5 and later.

6.8.65 NSImageNameApplicationIcon as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the imageNamed method.

The application's icon.

On versions of Mac OS X prior to v10.6, you can use the string "NSApplicationIcon".

Available in Mac OS X v10.6 and later.

6.8.66 `NSImageNameBluetoothTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A Bluetooth template image.

Available in Mac OS X v10.5 and later.

6.8.67 `NSImageNameBonjour` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A Bonjour icon.

Available in Mac OS X v10.5 and later.

6.8.68 `NSImageNameBookmarksTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

Bookmarks image suitable for a template.

Available in Mac OS X v10.6 and later.

6.8.69 `NSImageNameCaution` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

Caution Image.

Available in Mac OS X v10.6 and later.

6.8.70 `NSImageNameColorPanel` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A color panel toolbar icon.
Available in Mac OS X v10.5 and later.

6.8.71 NSImageNameColumnViewTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images used in segmented controls to switch the current view type.

Notes: To access this image, pass the specified constant to the imageNamed method.

A column view mode template image.

Available in Mac OS X v10.5 and later.

6.8.72 NSImageNameComputer as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Example:

```
dim n as NSImageMBS = NSImageMBS.imageNamed(NSImageMBS.NSImageNameComputer)
```

```
// set to size you need
n.size = NSMakeSizeMBS(512,512)
```

```
// and make picture
Backdrop = n.CopyPictureWithMask
```

Notes: To access this image, pass the specified constant to the imageNamed method.

A computer icon.

Available in Mac OS X v10.5 and later.

6.8.73 NSImageNameDotMac as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Notes: To access this image, pass the specified constant to the imageNamed method.

A Dot Mac icon.

Available in Mac OS X v10.5 and later.

6.8.74 `NSImageNameEnterFullScreenTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: An enter full-screen mode template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNamed` method.

6.8.75 `NSImageNameEveryone` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing sharing permission icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

Permissions for all users.

Available in Mac OS X v10.5 and later.

6.8.76 `NSImageNameExitFullScreenTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: An exit full-screen mode template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNamed` method.

6.8.77 `NSImageNameFlowViewTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images used in segmented controls to switch the current view type.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A cover flow view mode template image.

Available in Mac OS X v10.5 and later.

6.8.78 `NSImageNameFolder` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A folder image.

Available in Mac OS X v10.6 and later.

6.8.79 NSImageNameFolderBurnable as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Notes: To access this image, pass the specified constant to the imageNamed method.

A burnable folder icon.

Available in Mac OS X v10.5 and later.

6.8.80 NSImageNameFolderSmart as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Notes: A smart folder icon.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the imageNamed method.

6.8.81 NSImageNameFollowLinkFreestandingTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A link template image. You can use this image to implement a borderless button.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the imageNamed method.

6.8.82 NSImageNameFontPanel as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the imageNamed method.

A font panel toolbar icon.

Available in Mac OS X v10.5 and later.

6.8.83 `NSImageNameGoBackTemplate` as string

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

Function: The template name for the go back image.

Notes: Use these images for "go forward" or "go back" functions, as seen in Safari's toolbar. These images will automatically mirror when the user interface layout direction is right to left.

Available in macOS 10.12 and newer.

6.8.84 `NSImageNameGoForwardTemplate` as string

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

Function: The template name for the go forward image.

Notes: Use these images for "go forward" or "go back" functions, as seen in Safari's toolbar. These images will automatically mirror when the user interface layout direction is right to left.

Available in macOS 10.12 and newer.

6.8.85 `NSImageNameGoLeftTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A "go back" template image.

Available in Mac OS X v10.5 and later.

6.8.86 `NSImageNameGoRightTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A "go forward" template image.

Available in Mac OS X v10.5 and later.

6.8.87 `NSImageNameHomeTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

Home image suitable for a template.
Available in Mac OS X v10.6 and later.

6.8.88 NSImageNameIChatTheaterTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

An iChat Theater template image.

Available in Mac OS X v10.5 and later.

6.8.89 NSImageNameIconViewTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images used in segmented controls to switch the current view type.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

An icon view mode template image.

Available in Mac OS X v10.5 and later.

6.8.90 NSImageNameInfo as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

An information toolbar icon.

Available in Mac OS X v10.5 and later.

6.8.91 NSImageNameInvalidDataFreestandingTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

An invalid data template image. Place this icon to the right of any fields containing invalid data. You can use this image to implement a borderless button.

Available in Mac OS X v10.5 and later.

6.8.92 `NSImageNameLeftFacingTriangleTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A generic left-facing triangle template image.

Available in Mac OS X v10.5 and later.

6.8.93 `NSImageNameListViewTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images used in segmented controls to switch the current view type.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A list view mode template image.

Available in Mac OS X v10.5 and later.

6.8.94 `NSImageNameLockLockedTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A locked lock template image. Use to indicate locked content.

Available in Mac OS X v10.5 and later.

6.8.95 `NSImageNameLockUnlockedTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

An unlocked lock template image. Use to indicate modifiable content that can be locked.

Available in Mac OS X v10.5 and later.

6.8.96 `NSImageNameMenuMixedStateTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A horizontal dash. Drawing these outside of menus is discouraged.
Available in Mac OS X v10.6 and later.

6.8.97 NSImageNameMenuOnStateTemplate as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the imageNamed method.

A check mark. Drawing these outside of menus is discouraged.

Available in Mac OS X v10.6 and later.

6.8.98 NSImageNameMobileMe as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the imageNamed method.

MobileMe logo. Note that this is preferred to using the NSImageNameDotMac image, although that image is not expected to be deprecated.

Available in Mac OS X v10.6 and later.

6.8.99 NSImageNameMultipleDocuments as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Drag image you can use in your applications.

Notes: A drag image for multiple items.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the imageNamed method.

6.8.100 NSImageNameNetwork as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing Finder items.

Notes: A network icon.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the imageNamed method.

6.8.101 `NSImageNamePathTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

A path button template image.

Available in Mac OS X v10.5 and later.

6.8.102 `NSImageNamePreferencesGeneral` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

General preferences toolbar icon. Use in a preferences window only.

Available in Mac OS X v10.5 and later.

6.8.103 `NSImageNameQuickLookTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

A Quick Look template image.

Available in Mac OS X v10.5 and later.

6.8.104 `NSImageNameRefreshFreestandingTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A refresh template image. You can use this image to implement a borderless button.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNameNamed` method.

6.8.105 `NSImageNameRefreshTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A refresh template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNameNamed` method.

6.8.106 `NSImageNameRemoveTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Example:

```
Dim n As NSImageMBS = NSImageMBS.imageNamed(NSImageMBS.NSImageNameRemoveTemplate)
```

```
canvas1.backdrop = n.CopyPictureWithAlpha
```

Notes: A remove item template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNameNamed` method.

6.8.107 `NSImageNameRevealFreestandingTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A reveal contents template image. You can use this image to implement a borderless button.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNameNamed` method.

6.8.108 `NSImageNameRightFacingTriangleTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

A generic right-facing triangle template image.

Available in Mac OS X v10.5 and later.

6.8.109 `NSImageNameShareTemplate` as string

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: One of the images used in segmented controls to switch the current view type.

Notes: A share view template image.

Available in OS X v10.8 and later.

6.8.110 `NSImageNameSlideshowTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

A slideshow template image.

Available in Mac OS X v10.5 and later.

6.8.111 `NSImageNameSmartBadgeTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A badge for a "smart" item.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNameNamed` method.

6.8.112 `NSImageNameStatusAvailable` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

Small green indicator, similar to iChat's available image.

Available in Mac OS X v10.6 and later.

6.8.113 `NSImageNameStatusNone` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

Small clear indicator.

Available in Mac OS X v10.6 and later.

6.8.114 `NSImageNameStatusPartiallyAvailable` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

Small yellow indicator, similar to iChat's idle image.

Available in Mac OS X v10.6 and later.

6.8.115 `NSImageNameStatusUnavailable` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNameNamed` method.

Small red indicator, similar to iChat's unavailable image.

Available in Mac OS X v10.6 and later.

6.8.116 `NSImageNameStopProgressFreestandingTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

A stop progress template image. You can use this image to implement a borderless button.

Available in Mac OS X v10.5 and later.

6.8.117 `NSImageNameStopProgressTemplate` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing standard artwork and icons that you can use in your applications.

Notes: A stop progress button template image.

Available in Mac OS X v10.5 and later.

To access this image, pass the specified constant to the `imageNamed` method.

6.8.118 `NSImageNameTouchBarAddDetailTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAddDetailTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.119 `NSImageNameTouchBarAddTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAddTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.120 NSImageNameTouchBarAlarmTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAlarmTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.121 NSImageNameTouchBarAudioInputMuteTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioInputMuteTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.122 `NSImageNameTouchBarAudioInputTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioInputTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.123 `NSImageNameTouchBarAudioOutputMuteTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioOutputMuteTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.124 `NSImageNameTouchBarAudioOutputVolumeHighTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioOutputVolumeHighTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.125 NSImageNameTouchBarAudioOutputVolumeLowTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioOutputVolumeLowTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.126 NSImageNameTouchBarAudioOutputVolumeMediumTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioOutputVolumeMediumTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.127 `NSImageNameTouchBarAudioOutputVolumeOffTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarAudioOutputVolumeOffTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.128 `NSImageNameTouchBarBookmarksTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarBookmarksTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.129 NSImageNameTouchBarColorPickerFill as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarColorPickerFill
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.130 NSImageNameTouchBarColorPickerFont as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarColorPickerFont
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.131 NSImageNameTouchBarColorPickerStroke as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarColorPickerStroke
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.132 NSImageNameTouchBarCommunicationAudioTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarCommunicationAudioTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.133 NSImageNameTouchBarCommunicationVideoTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarCommunicationVideoTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.134 NSImageNameTouchBarComposeTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarComposeTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.135 NSImageNameTouchBarDeleteTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarDeleteTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.136 NSImageNameTouchBarDownloadTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarDownloadTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.137 NSImageNameTouchBarEnterFullScreenTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarEnterFullScreenTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.138 NSImageNameTouchBarExitFullScreenTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarExitFullScreenTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.139 NSImageNameTouchBarFastForwardTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarFastForwardTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.140 NSImageNameTouchBarFolderCopyToTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarFolderCopyToTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.141 NSImageNameTouchBarFolderMoveToTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarFolderMoveToTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.142 NSImageNameTouchBarFolderTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarFolderTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.143 NSImageNameTouchBarGetInfoTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarGetInfoTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.144 NSImageNameTouchBarGoBackTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarGoBackTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.145 NSImageNameTouchBarGoDownTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarGoDownTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.146 `NSImageNameTouchBarGoForwardTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarGoForwardTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.147 `NSImageNameTouchBarGoUpTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarGoUpTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.148 `NSImageNameTouchBarHistoryTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarHistoryTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.149 NSImageNameTouchBarIconViewTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarIconViewTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.150 NSImageNameTouchBarListViewTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarListViewTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.151 NSImageNameTouchBarMailTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarMailTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.152 NSImageNameTouchBarNewFolderTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarNewFolderTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.153 NSImageNameTouchBarNewMessageTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarNewMessageTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.154 NSImageNameTouchBarOpenInBrowserTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarOpenInBrowserTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.155 NSImageNameTouchBarPauseTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarPauseTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.156 `NSImageNameTouchBarPlayheadTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarPlayheadTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.157 `NSImageNameTouchBarPlayPauseTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarPlayPauseTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.158 `NSImageNameTouchBarPlayTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarPlayTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.159 NSImageNameTouchBarQuickLookTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarQuickLookTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.160 NSImageNameTouchBarRecordStartTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRecordStartTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.161 NSImageNameTouchBarRecordStopTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRecordStopTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.162 NSImageNameTouchBarRefreshTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRefreshTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.163 NSImageNameTouchBarRewindTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRewindTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.164 NSImageNameTouchBarRotateLeftTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRotateLeftTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.165 NSImageNameTouchBarRotateRightTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarRotateRightTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.166 NSImageNameTouchBarSearchTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarSearchTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.167 NSImageNameTouchBarShareTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarShareTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.168 NSImageNameTouchBarSidebarTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSidebarTemplate
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.169 NSImageNameTouchBarSkipAhead15SecondsTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSkipAhead15SecondsTemplate
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.170 NSImageNameTouchBarSkipAhead30SecondsTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarSkipAhead30SecondsTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.171 NSImageNameTouchBarSkipAheadTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarSkipAheadTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.172 NSImageNameTouchBarSkipBack15SecondsTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarSkipBack15SecondsTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.173 `NSImageNameTouchBarSkipBack30SecondsTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSkipBack30SecondsTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.174 `NSImageNameTouchBarSkipBackTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSkipBackTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.175 `NSImageNameTouchBarSkipToEndTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSkipToEndTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.176 NSImageNameTouchBarSkipToStartTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSkipToStartTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.177 NSImageNameTouchBarSlideshowTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarSlideshowTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.178 NSImageNameTouchBarTagIconTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTagIconTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.179 NSImageNameTouchBarTextBoldTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextBoldTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.180 `NSImageNameTouchBarTextBoxTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextBoxTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.181 `NSImageNameTouchBarTextCenterAlignTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextCenterAlignTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.182 `NSImageNameTouchBarTextItalicTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextItalicTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.183 NSImageNameTouchBarTextJustifiedAlignTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextJustifiedAlignTemplate
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.184 NSImageNameTouchBarTextLeftAlignTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextLeftAlignTemplate
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.185 `NSImageNameTouchBarTextListTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarTextListTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.186 `NSImageNameTouchBarTextRightAlignTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
dim img as NSImageMBS

name = NSImageMBS.NSImageNameTouchBarTextRightAlignTemplate
img = NSImageMBS.imageNamed(n)

Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.187 `NSImageNameTouchBarTextStrikethroughTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextStrikethroughTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.188 NSImageNameTouchBarTextUnderlineTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarTextUnderlineTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.189 NSImageNameTouchBarUserAddTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarUserAddTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.190 `NSImageNameTouchBarUserGroupTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarUserGroupTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.191 `NSImageNameTouchBarUserTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarUserTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.192 `NSImageNameTouchBarVolumeDownTemplate` as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarVolumeDownTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.193 NSImageNameTouchBarVolumeUpTemplate as string

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

Function: One of the touchbar image templates.

Example:

```
dim name as string
```

```
dim img as NSImageMBS
```

```
name = NSImageMBS.NSImageNameTouchBarVolumeUpTemplate
```

```
img = NSImageMBS.imageNamed(n)
```

```
Backdrop = img.CopyPictureWithAlpha
```

Notes: Available in macOS 10.12 and newer.

6.8.194 NSImageNameTrashEmpty as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the imageNamed method.

An image of the empty trash can.

Available in Mac OS X v10.6 and later.

6.8.195 `NSImageNameTrashFull` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.
An image of the full trash can.
Available in Mac OS X v10.6 and later.

6.8.196 `NSImageNameUser` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing sharing permission icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.
Permissions for a single user.
Available in Mac OS X v10.5 and later.

6.8.197 `NSImageNameUserAccounts` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.
User account toolbar icon. Use in a preferences window only.
Available in Mac OS X v10.5 and later.

6.8.198 `NSImageNameUserGroup` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images representing sharing permission icons that you can use in your applications.

Notes: To access this image, pass the specified constant to the `imageNamed` method.
Permissions for a group of users.
Available in Mac OS X v10.5 and later.

6.8.199 `NSImageNameUserGuest` as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: One of the images that you can use in application toolbars.

Notes: To access this image, pass the specified constant to the `imageNamed` method.

Shaded user figure.

Available in Mac OS X v10.6 and later.

6.8.200 PNGRepresentation as Memoryblock

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The image as the binary data in a PNG file.

Example:

```
dim img as NSImageMBS
dim p as Picture
dim f as FolderItem
dim b as BinaryStream

p=New Picture(100,100,32)
p.Graphics.ForeColor=&cFF0000
p.Graphics.FillOval 0,0,100,100
Backdrop=p
img=new NSImageMBS(p,p.Mask)

f=SpecialFolder.Desktop.Child("test.png")
b=f.CreateBinaryFile("")
b.Write img.PNGRepresentation
b.Close

f.Launch
```

Notes: PNG does support masks.

6.8.201 PNGRepresentationMT as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The image as the binary data in a PNG file.

Example:

```
// take some Picture
dim logo as Picture = LogoMBS(500)

// make a NSImageMBS from it
dim nsimage as new NSImageMBS(logo)

// use thread friendly compress function
```

```
dim jpeg as MemoryBlock = nsimage.PNGRepresentationMT
// decode to see if it worked
dim test as Picture = PNGStringToPictureMBS(jpeg)
// and display
Backdrop = test
```

Notes: PNG does support masks.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.202 recache

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Invalidates and frees the offscreen caches of all image representations.

Notes: If you modify an image representation, you must send a recache message to the corresponding image object to force the changes to be recached. The next time any image representation is drawn, it is asked to recreate its cached image. If you do not send this message, the image representation may use the old cache data. This method simply clears the cached image data; it does not delete the NSCachedImageRep objects associated with any image representations.

If you do not plan to use an image again right away, you can free its caches to reduce the amount of memory consumed by your program.

6.8.203 removeRepresentation(img as NSImageRepMBS)

Plugin Version: 8.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the specified image representation from the receiver and releases it.

6.8.204 RepresentationHeight(index as Integer) as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Height of the representation with the given index in pixel.

Notes: Index from 0 to RepresentationCount-1.

Returns 0 on invalid index.

6.8.205 representations as NSImageRepMBS()

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array containing all of the receiver's image representations.

Notes: An array containing zero or more NSImageRep objects.

Returns nil on any error.

6.8.206 RepresentationWidth(index as Integer) as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Width of the representation with the given index in pixel.

Notes: Index from 0 to RepresentationCount-1.

Returns 0 on invalid index.

6.8.207 setName(value as String) as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Registers the image under the specified name.

Notes: Returns true if the receiver was successfully registered with the given name; otherwise, false.

Discussion

If the receiver is already registered under a different name, this method unregisters the other name. If a different image is registered under the name specified in aString, this method does nothing and returns false.

When naming an image using this method, it is convention not to include filename extensions in the names you specify. That way, you can easily distinguish between images you have named explicitly and those you want to load from the application's bundle. For information about the rules used to search for images, see the `imageNamed` method.

6.8.208 `setSize(width as Double, height as Double)`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the width and height of the image.

Notes: Pass the new size of the image, measured in points.

Discussion:

The size of an `NSImage` object must be set before it can be used. If the size of the image hasn't already been set when an image representation is added, the size is taken from the image representation's data. For EPS images, the size is taken from the image's bounding box. For TIFF images, the size is taken from the `ImageLength` and `ImageWidth` attributes.

Changing the size of an `NSImage` after it has been used effectively resizes the image. Changing the size invalidates all its caches and frees them. When the image is next composited, the selected representation will draw itself in an offscreen window to recreate the cache.

6.8.209 `TIFFRepresentation as Memoryblock`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns a string containing TIFF data for all of the image representations in the receiver.

Example:

```
Dim p As Picture = LogoMBS(500)

// save as tiff

Dim n As New NSImageMBS(p)
Dim tiff As String = n.TIFFRepresentation
Dim file1 As FolderItem = SpecialFolder.Desktop.Child("test1.tif")
Dim b As BinaryStream = BinaryStream.Create(file1)
b.Write tiff
b.Close
```

Notes: A string containing the TIFF data, or nil if the TIFF data could not be created.

Discussion:

You can use the returned data object to write the TIFF data to a file. For each image representation, this method uses the TIFF compression option associated with that representation or NSTIFFCompressionNone, if no option is set.

If one of the receiver's image representations does not support the creation of TIFF data natively (PDF and EPS images, for example), this method creates the TIFF data from that representation's cached content.

6.8.210 TIFFRepresentationMT as Memoryblock

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns a string containing TIFF data for all of the image representations in the receiver.

Example:

```
// take some Picture
dim logo as Picture = LogoMBS(500)

// make a NSImageMBS from it
dim nsimage as new NSImageMBS(logo)

// use thread friendly compress function
dim data as MemoryBlock = nsimage.TIFFRepresentationMT

// decode to see if it worked
dim test as Picture = TIFFStringToPictureMBS(data)

// and display
Backdrop = test
```

Notes: A string containing the TIFF data, or nil if the TIFF data could not be created.

Discussion:

You can use the returned data object to write the TIFF data to a file. For each image representation, this method uses the TIFF compression option associated with that representation or NSTIFFCompressionNone, if no option is set.

If one of the receiver's image representations does not support the creation of TIFF data natively (PDF and EPS images, for example), this method creates the TIFF data from that representation's cached content.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.211 `TIFFRepresentationUsingCompression(comp as Integer, factor as Double) as Memoryblock`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns a string containing TIFF data for all of the image representations in the receiver.

Notes: `comp`: The type of compression to use. For a list of values, see the constants in `NSBitmapImageRep`.

`aFloat`: Provides a hint for compression types that implement variable compression ratios. Currently, only JPEG compression uses a compression factor.

Returns a string containing the TIFF data, or nil if the TIFF data could not be created.

Discussion:

You can use the returned data object to write the TIFF data to a file. If the specified compression isn't applicable, no compression is used. If a problem is encountered during generation of the TIFF data, this method may raise an exception.

If one of the receiver's image representations does not support the creation of TIFF data natively (PDF and EPS images, for example), this method creates the TIFF data from that representation's cached content.

6.8.212 `TIFFRepresentationUsingCompressionMT(comp as Integer, factor as Double) as Memoryblock`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns a string containing TIFF data for all of the image representations in the receiver.

Notes: `comp`: The type of compression to use. For a list of values, see the constants in `NSBitmapImageRep`.

`aFloat`: Provides a hint for compression types that implement variable compression ratios. Currently, only JPEG compression uses a compression factor.

Returns a string containing the TIFF data, or nil if the TIFF data could not be created.

Discussion:

You can use the returned data object to write the TIFF data to a file. If the specified compression isn't applicable, no compression is used. If a problem is encountered during generation of the TIFF data, this

method may raise an exception.

If one of the receiver's image representations does not support the creation of TIFF data natively (PDF and EPS images, for example), this method creates the TIFF data from that representation's cached content.

The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.

If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

6.8.213 Properties

6.8.214 accessibilityDescription as string

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: The image's accessibility description.

Notes: A short localized string that does not include the name of the interface element.

This description will be used automatically by interface elements that display images. Like all accessibility descriptions, the string should be a short localized string that does not include the name of the interface element. For instance, "delete" rather than "delete button".

(Read and Write property)

6.8.215 backgroundColor as NSColorMBS

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: The background color of image.

Notes: The background color of the image. The default color is transparent, as returned by the clearColor method of NSColor.

The background color is visible only if the drawn image representation does not completely cover all of the pixels available for the image's current size.

The value of this variant must be an object of class NSColorMBS.

(Read and Write property)

6.8.216 `cacheDepthMatchesImageDepth` as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether an image's offscreen window caches use the same bit depth as the image data itself.

Notes: Returns true if the offscreen window caches use the same bit depth as the image data; otherwise, false. The default value is false.

(Read and Write property)

6.8.217 `cacheMode` as Integer

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The receiver's caching mode.

Notes: The caching mode determines when the receiver's image representations use offscreen caches. Offscreen caches speed up rendering time but do so by using extra memory. In the default caching mode (`NSImageCacheDefault`), each image representation chooses the caching technique that produces the fastest drawing times. For example, in the default mode, the `NSPDFImageRep` and `NSEPSImageRep` classes use the `NSImageCacheAlways` mode but the `NSBitmapImageRep` class uses the `NSImageCacheBySize` mode.

(Read and Write property)

6.8.218 `EXIFData` as Dictionary

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns a dictionary with exif data.

Notes: (Read only property)

6.8.219 `Handle` as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The handle to the internally used `NSImage` object.

Notes: (Read and Write property)

6.8.220 `height` as Double

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The height of the image or zero.

Notes: If no size has been set and the size cannot be determined from any of the receiver's image represen-

tations.

(Read only property)

6.8.221 isCachedSeparately as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether each image representation caches its contents in a separate offscreen window.

Notes: Returns true if the image representations cache their content in separate offscreen windows; otherwise, false. The default value is false.

Discussion:

If this method returns false, it means that the image may be cached in a shared window but is not required to be. Images are cached in a shared window if they have the same general attributes, such as color space, resolution, and bit depth.

(Read and Write property)

6.8.222 isDataRetained as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether the receiver retains its source image data.

Notes: Returns true if the image retains its source data; otherwise, false. The default value is false with some exceptions, which are covered in the discussion.

(Read and Write property)

6.8.223 isFlipped as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether the image uses a flipped coordinate system.

Notes: Returns true if the image's coordinate system is flipped; otherwise, false. The default is false.

(Read and Write property)

6.8.224 isTemplate as Boolean

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the image is a template image.

Example:

```
Dim ActionGearTemplate as string // content of PDF file
Dim Image as New NSImageMBS( ActionGearTemplate )
Image.setSize( 14,14 )
Image.IsTemplate = true
```

Notes: Value is true if the image is a template image; otherwise, false.

Template images consist of black and clear colors (and an alpha channel). Template images are not intended to be used as standalone images and are usually mixed with other content to create the desired final appearance.

Available in Mac OS X v10.5 and later.

Assigning value to `istemplate`, will internally call `setTemplate`.

(Read and Write property)

6.8.225 isValid as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether the image can be drawn.

Notes: Return value: true if the receiver can be drawn; otherwise, false.

If the object is initialized with an existing image file, but the corresponding image data is not yet loaded into memory, this method loads the data and expands it as needed. If the object contains no image representations and no associated image file, this method creates a valid cached image representation and initializes it to the default bit depth. This method returns false in cases where the file or URL from which it was initialized is nonexistent or when the data in an existing file is invalid.

(Read only property)

6.8.226 matchesOnMultipleResolution as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether image representations whose resolution is an integral multiple of the device resolution are considered a match.

Notes: Returns true if image representations whose resolution is an integral multiple of the device resolution are considered a match; otherwise, false.

Discussion:

When this method returns false, only image representations whose resolution is exactly the same as the device resolution are considered matches. If this method returns true and multiple image representations fit this criteria, the one whose resolution is closest to the device resolution is chosen.

The default value is true.
(Read and Write property)

6.8.227 MaximumPixelHeight as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: From all the image representation the height of the heighest one.

Notes: Value is returned in pixels. Returns 0 on any error.

(Read only property)

6.8.228 MaximumPixelWidth as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: From all the image representation the width of the widest one.

Notes: Value is returned in pixels. Returns 0 on any error.

(Read only property)

6.8.229 MinimumPixelHeight as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: From all the image representation the width of the smallest one.

Notes: Value is returned in pixels. Returns 0 on any error.

(Read only property)

6.8.230 MinimumPixelWidth as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: From all the image representation the width of the smallest one.

Notes: Value is returned in pixels. Returns 0 on any error.

(Read only property)

6.8.231 name as String

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the name associated with the receiver, if any.

Notes: (Read only property)

6.8.232 prefersColorMatch as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns a Boolean value indicating whether the image prefers to choose image representations using color matching or resolution matching.

Notes: Discussion:

Both color matching and resolution matching may influence the choice of an image representation. This method simply indicates which technique is used first during the selection process. The default value is true. (Read and Write property)

6.8.233 RepresentationsCount as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Number of representations this image has.

Notes: Same as Representations.Count

(Read only property)

6.8.234 scalesWhenResized as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether image representations are scaled to fit the receiver's size.

Notes: Returns true if image representations are scaled to fit the receiver; otherwise, false. The default value is false.

Images are not resized during drawing if this method returns true. They are only resized when you change the size by sending the receiver a setSize message.

(Read and Write property)

6.8.235 size as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The size of the receiver.

Notes: The size of the receiver or (0.0, 0.0) if no size has been set and the size cannot be determined from

any of the receiver's image representations.

The size of an `NSImage` object must be set before it can be used. If the size of the image hasn't already been set when an image representation is added, the size is taken from the image representation's data. For EPS images, the size is taken from the image's bounding box. For TIFF images, the size is taken from the `ImageLength` and `ImageWidth` attributes.

Changing the size of an `NSImage` after it has been used effectively resizes the image. Changing the size invalidates all its caches and frees them. When the image is next composited, the selected representation will draw itself in an offscreen window to recreate the cache.

(Read and Write property)

6.8.236 `symbolConfiguration` as `NSImageSymbolConfigurationMBS`

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: The configuration details for a symbol image.

Notes: Use this property to access the traits and rendering attributes the system uses with the symbol image. These details determine which variant of the image to load and draw and how to render it, falling back on the current environment for values that you don't specify. For symbol images, the default value of this property is a symbol image configuration object with unspecified values.

You can't modify this property directly, but you can use `imageWithSymbolConfiguration()` when you want to create a new image object with a specific set of traits.

For macOS 12.0 or newer.

(Read only property)

6.8.237 `usesEPSOnResolutionMismatch` as `Boolean`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether EPS image representations are preferred when no other representations match the resolution of the device.

Notes: (Read and Write property)

6.8.238 `width` as `Double`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The width of the image or zero.

Notes: If no size has been set and the size cannot be determined from any of the receiver's image represen-

tations.

(Read only property)

6.8.239 Constants

Image Cache Modes

Constant	Value	Description
<code>NSImageCacheAlways</code>	1	Always generate a cache when drawing.
<code>NSImageCacheBySize</code>	2	Cache if cache size is smaller than the original data.
<code>NSImageCacheDefault</code>	0	Caching is unspecified. Use the image rep's default.
<code>NSImageCacheNever</code>	3	Never cache; always draw direct.

Image Load Status

Constant	Value	Description
<code>NSImageLoadStatusCancelled</code>	1	Image loading was canceled. The image contains the portions of the data that have already been successfully decompressed, if any.
<code>NSImageLoadStatusCompleted</code>	0	Enough data has been provided to completely decompress the image.
<code>NSImageLoadStatusInvalidData</code>	2	An error occurred during image decompression. The image data is probably corrupt. The image contains the portions of data that have already been successfully decompressed, if any.
<code>NSImageLoadStatusReadError</code>	4	Not enough data was available for full decompression of the image. The image contains the portions of the data that have already been successfully decompressed, if any.
<code>NSImageLoadStatusUnexpectedEOF</code>	3	Not enough data was available for full decompression of the image. The image contains the portions of the data that have already been successfully decompressed, if any.

6.9 class NSImageRepMBS

6.9.1 class NSImageRepMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSImageRep class is a semiabstract superclass (“semi” because it has some instance variables and implementation of its own).

Notes: Each of its subclasses knows how to draw an image from a particular kind of source data. While an NSImageRep subclass can be used directly, it is typically through an NSImage object. An NSImage object manages a group of image representations, choosing the best one for the current output device.

Subclasses are not implemented in the plugin currently.

All methods in this class will catch exceptions from Cocoa and raise a NSExcptionMBS instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

This is an abstract class. You can’t create an instance, but you can get one from various plugin functions.

6.9.2 Methods

6.9.3 canInitWithData(data as memoryblock) as Boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether the receiver can initialize itself from the specified data.

Notes: Returns true if the receiver understands the format of the specified data and can use it to initialize itself; otherwise, false.

This method should be overridden by subclasses. Note that this method does not need to do a comprehensive check of the image data; it should return false only if it knows it cannot initialize itself from the data.

6.9.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

6.9.5 setSize(width as Double, height as Double)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the size of the image representation to the specified value.

Notes: This method determines the size of the image when it's rendered. It is not necessarily the same as the width and height of the image in pixels as specified by the image data, nor must it be equal to the size set for the `NSImage` object that wraps this image representation. You must set the image size before you can render it.

The size of an image representation combined with the physical dimensions of the image data determine the resolution of the image.

6.9.6 Properties

6.9.7 bitsPerSample as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of bits per sample in the receiver.

Notes: (Read and Write property)

6.9.8 colorSpaceName as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The name of the receiver's color space.

Notes: The colorspace name, or `NSCalibratedRGBColorSpace` if no name has been assigned.

By default, an `NSImageRep` object's color space name is `NSCalibratedRGBColorSpace`. Color space names are defined as part of the `NSColor` class, in `NSGraphics.h`. The following are valid color space names:

`NSCalibratedWhiteColorSpace`
`NSCalibratedBlackColorSpace`
`NSCalibratedRGBColorSpace`
`NSDeviceWhiteColorSpace`
`NSDeviceBlackColorSpace`
`NSDeviceRGBColorSpace`
`NSDeviceCMYKColorSpace`
`NSNamedColorSpace`
`NSCustomColorSpace`
(Read and Write property)

6.9.9 Handle as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

6.9.10 hasAlpha as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the receiver has an alpha channel.

Notes: Returns true if the receiver has a known alpha channel; otherwise, false.

Subclasses should call this method when loading image data to notify the parent class whether that data contains an alpha component. Passing in a value of true does not add an alpha channel to the image data itself; it merely records the fact that the data has an alpha channel.

(Read and Write property)

6.9.11 height as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Height of image in points.

Notes: 0 on any error.

(Read only property)

6.9.12 isOpaque as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a Boolean value indicating whether the receiver is opaque.

Notes: Use this method to test whether an image representation completely covers the area within the rectangle returned by the size method.

The returned value does not indicate whether the image has an alpha channel or if there is partial or complete transparency when drawing the image rep. Use the hasAlpha method to determine if the image has an alpha channel.

(Read and Write property)

6.9.13 pixelsHigh as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the height of the image, measured in pixels.

Notes: Return Value:

The height of the image, measured in the units of the device coordinate space. This value is usually derived from the image data itself.

(Read and Write property)

6.9.14 pixelsWide as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the width of the image, measured in pixels.

Notes: Return Value:

The width of the image, measured in the units of the device coordinate space. This value is usually derived from the image data itself.

(Read and Write property)

6.9.15 size as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: The size of the image representation.

Notes: This size is the size of the image representation when it's rendered. It is not necessarily the same as the width and height of the image in pixels as specified by the image data, nor must it be equal to the size set for the NSImage object that wraps this image representation.

The size of an image representation combined with the physical dimensions of the image data determine the resolution of the image.

(Read and Write property)

6.9.16 width as Double

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Width of image in points.

Notes: 0 on any error.

(Read only property)

6.9.17 Constants

Constants

Constant	Value	Description
NSImageRepMatchesDevice	0	One of the constants used by NSImageRep to denote an attribute whose value changes to match the display device. Indicates that the value of certain attributes, such as the number of colors or bits per sample, will change to match the display device. This value can be passed in (or received back) as the value of bitsPerSample, pixelsWide, and pixelsHigh.

6.10 class `NSImageSymbolConfigurationMBS`

6.10.1 class `NSImageSymbolConfigurationMBS`

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: An object that contains the specific font, style, and weight attributes to apply to a symbol image.

Example:

```
Dim config As NSImageSymbolConfigurationMBS = NSImageSymbolConfigurationMBS.configurationWith-  
PointSize(40, 0)
```

```
Dim n As NSImageMBS = NSImageMBS.imageWithSystemSymbolName("trash")
```

```
Dim o As NSImageMBS = n.imageWithSymbolConfiguration(config)
```

```
Backdrop = o.CopyPictureWithAlpha
```

Notes: Symbol image configuration objects include details such as the point size, scale, text style, and weight to apply to your symbol image. The system uses these details to determine which variant of the image to use and how to scale or style the image.

`NSImageSymbolConfiguration` objects are immutable after you create them. If you use the `configurationByApplyingConfiguration` method on the object, the new image attributes replace any previous attributes you supplied. After creating a symbol configuration object, assign it to the `symbolConfiguration` property of the `NSImageViewMBS` object you use to display the image. If you draw the image directly, use the `imageWithSymbolConfiguration` method to create a new image that contains the new attributes.

For design guidance, see Human Interface Guidelines.

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 22.1](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.1](#)
- [MBS Xojo Plugins, version 22.1pr1](#)

Xojo Developer Magazine

- [20.4, page 10: News](#)
- [20.3, page 10: News](#)

6.10.2 Methods

6.10.3 available as Boolean

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Whether the class is available.

Notes: Returns true on macOS 11.0 or later.

6.10.4 configurationByApplyingConfiguration(configuration as NSImageSymbolConfigurationMBS) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a configuration object by applying the values from the configuration you specify.

Notes: for macOS 12.0 or newer.

6.10.5 configurationPreferringMulticolor as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a configuration that specifies that the symbol should prefer its multicolor variant if one exists.

Notes: You can combine this configuration with one of the palette-based configurations. In that case, the symbol uses the multicolor variant if one exists; otherwise the symbol uses the palette version.

If the symbol supports neither, the symbol uses the monochrome (templated) symbol.

6.10.6 configurationWithHierarchicalColor(hierarchicalColor as NSColorMBS) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a hierarchical color configuration using the color you specify.

Notes: This method creates a color scheme based on a single color. The system reduces the intensity of the base color to create the secondary and tertiary colors.

When combining this with another configuration, the last configuration overrides existing values.

If the symbol doesn't have a palette variant, this color configuration doesn't have an effect, so the symbol uses the monochrome (templated) symbol.

6.10.7 configurationWithPaletteColors(paletteColors() as NSColorMBS) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a color configuration by specifying a palette of colors.

Example:

```
// shows red colored trash can
Dim colors() As NSColorMBS
colors.Append NSColorMBS.redColor
Dim config As NSImageSymbolConfigurationMBS = NSImageSymbolConfigurationMBS.configurationWith-
PaletteColors(colors)
Dim n As NSImageMBS = NSImageMBS.imageWithSystemSymbolName("trash")
Dim o As NSImageMBS = n.imageWithSymbolConfiguration(config)

Backdrop = o.CopyPictureWithAlpha
```

Notes: The system applies the colors sequentially per layer—the first color for the first layer, and the second color for the second layer. This is independent of the hierarchy level of the layer.

When you combine this with another configuration to create a palette, the last configuration overrides any existing color configuration.

If the symbol doesn't have a palette variant, this color configuration doesn't have an effect, so the symbol uses the monochrome (templated) symbol.

6.10.8 configurationWithPointSize(pointSize as Double, weight as Double) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a symbol configuration with the specified point size and font weight.

See also:

- 6.10.9 configurationWithPointSize(pointSize as Double, weight as Double, SymbolScale as Integer) as NSImageSymbolConfigurationMBS 754

6.10.9 configurationWithPointSize(pointSize as Double, weight as Double, SymbolScale as Integer) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a symbol configuration with the specified point size, font weight, and symbol scale.

See also:

- 6.10.8 configurationWithPointSize(pointSize as Double, weight as Double) as NSImageSymbolConfigurationMBS 754

6.10.10 configurationWithScale(SymbolScale as Integer) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a symbol configuration using the scale you specify.

Example:

```
Dim config As NSImageSymbolConfigurationMBS = NSImageSymbolConfigurationMBS.configurationWithScale(NSImageSymbolConfigurationMBS.kSymbolScaleLarge)
Dim n As NSImageMBS = NSImageMBS.imageWithSystemSymbolName("trash")
Dim o As NSImageMBS = n.imageWithSymbolConfiguration(config)
```

```
Backdrop = o.CopyPictureWithAlpha
```

6.10.11 configurationWithTextStyle(TextStyle as Integer) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a symbol configuration with the specified text style.

See also:

- 6.10.12 configurationWithTextStyle(TextStyle as Integer, SymbolScale as Integer) as NSImageSymbolConfigurationMBS 755

6.10.12 configurationWithTextStyle(TextStyle as Integer, SymbolScale as Integer) as NSImageSymbolConfigurationMBS

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a symbol configuration with the specified text style and symbol scale.

See also:

- 6.10.11 configurationWithTextStyle(TextStyle as Integer) as NSImageSymbolConfigurationMBS 755

6.10.13 Constructor

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: The private constructor.

6.10.14 NSFontTextStyleBody as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for body text.

6.10.15 NSFontTextStyleCallout as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for callouts.

6.10.16 NSFontTextStyleCaption1 as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for standard captions.

6.10.17 NSFontTextStyleCaption2 as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for alternate captions.

6.10.18 NSFontTextStyleFootnote as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use in footnotes.

6.10.19 NSFontTextStyleHeadline as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for headings.

6.10.20 NSFontTextStyleLargeTitle as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for titles.

6.10.21 NSFontTextStyleSubheadline as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for subheadings.

6.10.22 NSFontTextStyleTitle1 as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for first-level hierarchical headings.

6.10.23 NSFontTextStyleTitle2 as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for second-level hierarchical headings.

6.10.24 NSFontTextStyleTitle3 as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: One of the text style constants.

Notes: The font you use for third-level hierarchical headings.

6.10.25 Properties

6.10.26 Description as String

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: Shows description for debugging.

Example:

```
Dim config As NSImageSymbolConfigurationMBS = NSImageSymbolConfigurationMBS.configurationWithScale(NSImageSymbolConfigurationMBS.kSymbolScaleLarge)
```

```
MessageBox config.Description
```

Notes: (Read only property)

6.10.27 Handle as Integer

Plugin Version: 22.1, Platforms: macOS, iOS, Targets: All.

Function: The internal object reference number.

Notes: (Read only property)

6.10.28 Constants

Constants

Constant	Value	Description
kSymbolScaleLarge	3	Specifies which scale variant of a symbol image to use. The symbol uses the large scale variant. This scale is approximately 30% larger than the kSymbolScaleMedium scale.
kSymbolScaleMedium	2	Specifies which scale variant of a symbol image to use. The symbol uses the default medium scale variant.
kSymbolScaleSmall	1	Specifies which scale variant of a symbol image to use. The symbol uses the small scale variant. This scale is approximately 20% smaller than the kSymbolScaleMedium scale.

6.11 class NSPDFImageRepMBS

6.11.1 class NSPDFImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: An NSPDFImageRep object can render an image from a PDF format data stream.

Notes: Subclass of the NSImageRepMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.0pr5](#)

6.11.2 Methods

6.11.3 Constructor(data as Memoryblock)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSPDFImageRep object initialized with the specified PDF data.

Notes: Initialized NSPDFImageRep object. Initialization may fail if the PDF data does not conform to the PDF file format.

On success handle property is not zero.

6.11.4 imageRepWithData(data as Memoryblock) as NSPDFImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSPDFImageRep object initialized with the specified PDF data.

Notes: Returns an initialized NSPDFImageRep object or nil if the object could not be initialized. Initialization may fail if the PDF data does not conform to the PDF file format.

6.11.5 PDFRepresentation as Memoryblock

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the PDF representation of the receiver's image.

6.11.6 Properties

6.11.7 bounds as NSRectMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's bounding rectangle.

Notes: This value is equivalent to the crop box specified by the PDF data.
(Read only property)

6.11.8 currentPage as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The page currently displayed by the image representation.

Notes: A zero-based index indicating the page being displayed.
(Read and Write property)

6.11.9 pageCount as Integer

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the number of pages in the receiver.

Notes: (Read only property)

6.12 class NSPICTImageRepMBS

6.12.1 class NSPICTImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: An NSPICTImageRep object renders an image from a PICT format data stream as described in the Carbon QuickDraw Manager documentation.

Notes: This class can render PICT format version 1, version 2, and extended version 2 pictures.

Subclass of the NSImageRepMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.0pr5](#)

6.12.2 Methods

6.12.3 Constructor(data as Memoryblock)

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSPICTImageRep object initialized with the specified data.

Notes: data: A data object containing the PICT data.

Initialization may fail if the data does not conform to the PICT file format.

On success the handle property is not zero.

If the PICT data is obtained directly from a PICT file or document, this method ignores most of the 512-byte header that occurs before the start of the actual picture data. It may retrieve some relevant meta information from the header.

6.12.4 imageRepWithData(data as Memoryblock) as NSPICTImageRepMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates and returns an NSPICTImageRep object initialized with the specified data.

Notes: Initialization may fail if the data does not conform to the PICT file format.

On failure the handle property is nil.

6.12.5 PICTRepresentation as Memoryblock

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the receiver's PICT data.

Notes: A data object containing the PICT data. The returned data does not include the 512-byte header, if it was present in the original data. If you want to write the returned data to a file, you must precede it with a 512-byte header (containing all zeros) if you want to conform to the PICT document format.

6.12.6 Properties

6.12.7 boundingBox as NSRectMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the rectangle that bounds the receiver.

Notes: The rectangle bounding the receiver. This rectangle is obtained from the the picFrame field in the picture header. See the Carbon QuickDraw Manager documentation for information on the picture header. (Read only property)

Chapter 7

Cocoa Menus

7.1 class NSMenuItemMBS

7.1.1 class NSMenuItemMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: A class to represent a menuitem from the Cocoa world.

Notes: All methods in this class will catch exceptions from Cocoa and raise a NSExcptionMBS instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5](#)
- [More Menubar options in macOS Sonoma](#)
- [MBS Xojo Plugins, version 23.5pr2](#)
- [News from the MBS Xojo Plugins Version 23.3](#)
- [MBS Xojo Plugins, version 21.6pr3](#)
- [MBS Xojo Plugins 18.3](#)
- [Custom menu checkmarks](#)
- [Apply fonts to font PopupMenu](#)
- [MonkeyBread Software Releases the MBS Real Studio plug-ins in version 13.1](#)

Xojo Developer Magazine

- 5.6, page 49: Made in the Shade, Using Core Graphics in a REALbasic Project by Toby Rush
- 5.6, page 33: Third Party Plugins: Statusitems, Getting an icon in the top right of the Mac OS X menubar by Christian Schmitz
- 22.1, page 9: News
- 16.5, page 9: News

7.1.2 Methods

7.1.3 clearAction

Plugin Version: 20.3, Platform: macOS, Targets: Desktop only.

Function: Clears action event.

Notes: Call this method once to have a menuitem which has no action.

7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem)

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

Function: Creates a new NSMenuItemMBS object for a given MenuItem.

Notes: May raise UnsupportedOperationException or NilObjectException if used incorrectly.

See also:

- 7.1.5 Constructor(Handle as Integer) 764
- 7.1.6 Constructor(MenuItem as MenuItem) 765
- 7.1.7 Constructor(Other as NSMenuItemMBS) 766
- 7.1.8 Constructor(title as string="", keyEquivalent as string="") 766

7.1.5 Constructor(Handle as Integer)

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Constructor for creating an instanced based on an existing handle.

Example:

```
dim m as MenuItem = EditCopy
dim h as Integer = m.Handle(MenuItem.HandleType.CocoaNSMenuItem)
dim i as new NSMenuItemMBS(h)
```

```
i.Title = "Hello"
```

Notes: Useful if you get a NSMenuItem reference from a declare.
The object is retained.
See also:

- 7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem) 764
- 7.1.6 Constructor(MenuItem as MenuItem) 765
- 7.1.7 Constructor(Other as NSMenuItemMBS) 766
- 7.1.8 Constructor(title as string="", keyEquivalent as string="") 766

7.1.6 Constructor(MenuItem as MenuItem)

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new NSMenuItemMBS object for a given MenuItem.

Example:

```
dim FileDot, FileDiamond as MenuItem
```

```
FileDot.enable  
FileDot.checked = true
```

```
// via constructor  
dim n1 as new NSMenuItemMBS(fileDot)  
n1.onStateImage = NSImageMBS.imageNamed("NSMenuItemBullet")
```

```
FileDiamond.enable  
FileDiamond.checked = true
```

```
// via function  
dim n2 as NSMenuItemMBS = NSMenuItemMBS.MenuItem(FileDiamond)  
n2.onStateImage = NSImageMBS.imageNamed("NSMenuItemDiamond")
```

Notes: May raise UnsupportedOperationException or NilObjectException if used incorrectly.
See also:

- 7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem) 764
- 7.1.5 Constructor(Handle as Integer) 764
- 7.1.7 Constructor(Other as NSMenuItemMBS) 766
- 7.1.8 Constructor(title as string="", keyEquivalent as string="") 766

7.1.7 Constructor(Other as NSMenuItemMBS)

Plugin Version: 22.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new NSMenuItemMBS object referencing same NSMenuItem with a new Xojo object.
See also:

- 7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem) 764
- 7.1.5 Constructor(Handle as Integer) 764
- 7.1.6 Constructor(MenuItem as MenuItem) 765
- 7.1.8 Constructor(title as string="", keyEquivalent as string="") 766

7.1.8 Constructor(title as string="", keyEquivalent as string="")

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new menu items.

Notes: Handle is not 0 after this call if it was successful.
name and charcode are optional and can be "".

Charcode is the initial keyEquivalent for this menu item.

You can set the KeyEquivalentModifierMask to get different modifier keys.

See also:

- 7.1.4 Constructor(DesktopMenuItem as DesktopMenuItem) 764
- 7.1.5 Constructor(Handle as Integer) 764
- 7.1.6 Constructor(MenuItem as MenuItem) 765
- 7.1.7 Constructor(Other as NSMenuItemMBS) 766

7.1.9 CreateMenuItem(title as string="", keyEquivalent as string="")

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new menu items.

Notes: Handle is not 0 after this call if it was successful.
name and charcode are optional and can be "".

Charcode is the initial keyEquivalent for this menu item.

You can set the KeyEquivalentModifierMask to get different modifier keys.

Deprecated, please use the Constructor instead.

7.1.10 CreateSeparator

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Creates a Separator menu item.

7.1.11 MenuItem(DesktopMenuItem as DesktopMenuItem) as NSMenuItemMBS

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

Function: Queries NSMenuItemMBS for a given menu item.

Notes: Returns nil on any error.

See also:

- 7.1.12 MenuItem(MenuItem as MenuItem) as NSMenuItemMBS

767

7.1.12 MenuItem(MenuItem as MenuItem) as NSMenuItemMBS

Plugin Version: 18.3, Platform: macOS, Targets: Desktop only.

Function: Queries NSMenuItemMBS for a given menu item.

Example:

```
dim FileDot, FileDiamond as MenuItem
```

```
FileDot.enable
FileDot.checked = true
```

```
// via constructor
dim n1 as new NSMenuItemMBS(fileDot)
n1.onStateImage = NSImageMBS.imageNamed("NSMenuItemBullet")
```

```
FileDiamond.enable
FileDiamond.checked = true
```

```
// via function
dim n2 as NSMenuItemMBS = NSMenuItemMBS.MenuItem(FileDiamond)
n2.onStateImage = NSImageMBS.imageNamed("NSMenuItemDiamond")
```

Notes: Returns nil on any error.

See also:

- 7.1.11 MenuItem(DesktopMenuItem as DesktopMenuItem) as NSMenuItemMBS

767

7.1.13 sectionHeaderWithTitle(Title as String) as NSMenuItemMBS

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: Creates a menu item representing a section header with the provided title.

Example:

```
dim m as NSMenuItemMBS = NSMenuItemMBS.MenuItem(EditMenu)

// add the header
dim h as NSMenuItemMBS = NSMenuItemMBS.sectionHeaderWithTitle("Section Header")
h.Enabled = true
m.submenu.addItem h
self.SectionHeader = h

// and the new entry with our MyNSMenuItemMBS class to handle validate and action.
dim n as new MyNSMenuItemMBS("Hello")
n.Enabled = true
m.submenu.addItem n
self.HelloMenu = n // keep references
```

Notes: Section header items are used to provide context to a grouping of menu items. Items created using this method are non-interactive and do not perform an action.

Requires macOS Sonoma or newer.

7.1.14 separatorItem as NSMenuItemMBS

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Creates a new separator menu item.

7.1.15 setAction(target as NSResponderMBS, selectorName as string)

Plugin Version: 23.3, Platform: macOS, Targets: Desktop only.

Function: Sets the target for the menu action.

Example:

```
dim n as NSMenuItemMBS // your menu item

n.setAction(window1.NSWindowMBS, "performClose:")
```

Notes: Please make sure the selector is correct as otherwise you get an exception later.

And make sure the target object stays alive.

7.1.16 setTitleWithMnemonic(title as String)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets the title of a menu item with a character denoting an access key.

Notes: Use an ampersand character to mark the character (the one following the ampersand) to be designated.

Deprecated by Apple.

7.1.17 Properties

7.1.18 ActionSelector as String

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the name of the objective-c method called for this menu event.

Notes: You can use this to find menu items by their selector. Which is often more save than by index or title.

(Read only property)

7.1.19 Alternate as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether this menu item is an alternate to the previous menu item.

Notes: Available in Mac OS X v10.3 and later.

(Read and Write property)

7.1.20 attributedTitle as NSAttributedStringMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Specifies a custom string for a menu item.

Notes: You can use this method to add styled text and embedded images to menu item strings. If you do not set a text color for the attributed string, it is black when not selected, white when selected, and gray when disabled. Colored text remains unchanged when selected.

When you call this method to set the menu title to an attributed string, the setTitle: method is also called to set the menu title with a plain string. If you clear the attributed title, the plain title remains unchanged.

(Read and Write property)

7.1.21 Enabled as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Whether this menuItem is enabled.

Notes: (Read and Write property)

7.1.22 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The reference to the NSMenuItem object used internally.

Notes: (Read and Write property)

7.1.23 hasSubmenu as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns true if the menuItem has a submenu, false if it doesn't.

Notes: (Read only property)

7.1.24 Identifier as String

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: A string that identifies this user interface item.

Notes: It should be set to a unique value on NSViews when they are intended to be used inside a view-based NSTableView. Identifiers should be unique per-window. For programmatically created user interface items, you would typically set this value in code after creating a view but before adding it to a window. You may also want to set an identifier on a window, after creating it programmatically, to identify the window easily when it is reopened. You should not change the identifier after a view is added to a window. Identifiers beginning with an underscore are reserved for the system. In framework classes that implement this protocol, the accessor methods are not intended to be overridden.

To help avoid collision of identifiers, it is recommended that identifiers use the same prefix as is used for the framework or application. For example, identifiers for standard AppKit interface items, such as the open panel, will begin with "NS".

The slash '/', backslash '\', and colon ':' characters are reserved and should not be used in identifiers.
(Read and Write property)

7.1.25 image as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The image displayed by the menuitem, or nil if it displays no image.

Notes: (Read and Write property)

7.1.26 indentationLevel as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The menu item indentation level for the menu item.

Notes: The value will be from 0 to 15. The default indentation level is 0.

Available in Mac OS X v10.3 and later.

(Read and Write property)

7.1.27 isHidden as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the menuitem is hidden.

Notes: Mac OS X 10.5 only.

Returns true if the receiver is hidden, otherwise false.

This value can be set to hide or show a menuitem.

(Read and Write property)

7.1.28 isHiddenOrHasHiddenAncestor as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the menuitem or any of its superitems is hidden.

Notes: Mac OS X 10.5 only.

Returns true if the receiver or any of its superitems is hidden, otherwise false.

(Read only property)

7.1.29 isHighlighted as boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver should be drawn highlighted.

Notes: Returns true if the receiver should be drawn highlighted, otherwise false.

Mac OS X 10.5 only.

(Read only property)

7.1.30 isSectionHeader as Boolean

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: Indicates whether the item is a section header.

Notes: Section header items are created using the `sectionHeader()` class method.

Requires macOS Sonoma or newer.

(Read only property)

7.1.31 isSeparatorItem as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns whether the receiver is a separator item (that is, a menu item used to visually segregate related menu items).

Notes: (Read only property)

7.1.32 keyEquivalent as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The receiver's unmodified keyboard equivalent, or the empty string if one hasn't been defined.

Example:

```
dim m as new NSMenuItemMBS
m.CreateMenuItem "Hello"
m.keyEquivalent = "a" // A
m.keyEquivalent = "A" // shift-A
```

Notes: Use `keyEquivalentModifierMask` to determine the modifier mask for the key equivalent. In the current implementation "A" can be interpreted as Shift-A by the system even without the shift in the mask.

(Read and Write property)

7.1.33 keyEquivalentModifierMask as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: the menu item's keyboard equivalent modifier mask.

Example:

```
const NSShiftKeyMask=131072
const NSControlKeyMask=262144
const NSAlternateKeyMask=524288
const NSCommandKeyMask=1048576

dim d as new NSMenuItemMBS
d.CreateMenuItem "Last menu entry", ""
d.Enabled=true
d.KeyEquivalent="A"
d.KeyEquivalentModifierMask=NSShiftKeyMask+NSCommandKeyMask+NSAlternateKeyMask // command-
option-shift
```

Notes: Constants for the mask:

NSAlphaShiftKeyMask = 65536
Set if Caps Lock key is pressed.
Available in Mac OS X v10.0 and later.

NSShiftKeyMask = 131072
Set if Shift key is pressed.
Available in Mac OS X v10.0 and later.

NSControlKeyMask = 262144
Set if Control key is pressed.
Available in Mac OS X v10.0 and later.

NSAlternateKeyMask = 524288
Set if Option or Alternate key is pressed.
Available in Mac OS X v10.0 and later.

NSCommandKeyMask = 1048576
Set if Command key is pressed.
Available in Mac OS X v10.0 and later.

`NSNumericPadKeyMask = 2097152`

Set if any key in the numeric keypad is pressed. The numeric keypad is generally on the right side of the keyboard. This is also set if any of the arrow keys are pressed (`NSUpArrowFunctionKey`, `NSDownArrowFunctionKey`, `NSLeftArrowFunctionKey`, and `NSRightArrowFunctionKey`).

Available in Mac OS X v10.0 and later.

`NSHelpKeyMask = 4194304`

Set if the Help key is pressed.

Available in Mac OS X v10.0 and later.

`NSFunctionKeyMask = 8388608`

Set if any function key is pressed. The function keys include the F keys at the top of most keyboards (F1, F2, and so on) and the navigation keys in the center of most keyboards (Help, Forward Delete, Home, End, Page Up, Page Down, and the arrow keys).

Available in Mac OS X v10.0 and later.

`NSDeviceIndependentModifierFlagsMask = 16777216`

Used to retrieve only the device-independent modifier flags, allowing applications to mask off the device-dependent modifier flags, including event coalescing information.

Available in Mac OS X v10.4.

(Read and Write property)

7.1.34 `menu` as `NSMenuMBS`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The menu where this menu item is inside.

Notes: nil if no menu belongs to this menuitem.

(Read only property)

7.1.35 `mixedStateImage` as `NSImageMBS`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The image used to depict a "mixed state."

Notes: A mixed state is useful for indicating "off" and "on" attribute values in a group of selected objects, such as a selection of text containing bold and plain (nonbolded) words.

(Read and Write property)

7.1.36 offStateImage as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The image used to depict the receiver's "off" state, or nil if the image has not been set.

Notes: By default, there is no off state image.

(Read and Write property)

7.1.37 onStateImage as NSImageMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The image of the receiver that indicates an "on" state.

Notes: The Image object to use for the "on" state of the menu item. If itemImage is nil, any current on-state image is removed.

(Read and Write property)

7.1.38 parentItem as NSMenuItemMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the menu item whose submenu contains the receiver.

Notes: Returns the parent menu item, or nil if the receiver does not have a parent item.

Available in Mac OS X v10.6 and later.

(Read only property)

7.1.39 state as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The state of the menuitem.

Notes: An integer constant representing a state; it should be one of NSOffState, NSOnState, or NSMixedState.

Same as Checked property on Xojo's menu items.

Constants:

NSMixedState = -1	The corresponding feature is in effect somewhere.
NSOffState = 0	The corresponding feature is in effect nowhere.
NSOnState = 1	The corresponding feature is in effect everywhere.

(Read and Write property)

7.1.40 submenu as NSMenuItemMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The submenu attached to this menuitem.

Notes: nil if there is no submenu.

(Read and Write property)

7.1.41 tag as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The menu item tag value.

Notes: You can use this value as you like.

If you need to store more custom data than just an integer, you should subclass the NSMenuItemMBS class and add properties as needed.

(Read and Write property)

7.1.42 Title as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The title of the menu item.

Notes: (Read and Write property)

7.1.43 toolTip as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The help tag for a menu item.

Notes: (Read and Write property)

7.1.44 userKeyEquivalent as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The user-assigned key equivalent for the menu item.

Notes: (Read only property)

7.1.45 view as NSViewMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The view to be used for this menuitem.

Notes: Using a view you can draw whatever you like in the menu item.

You can set it to nil to remove the view.

Requires Mac OS X 10.5.

(Read and Write property)

7.1.46 usesUserKeyEquivalents as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop only.

Function: Whether menu items conform to user preferences for key equivalents.

Notes: If true, menu items conform to user preferences for key equivalents; if false, the key equivalents originally assigned to the menu items are used.

(Read and Write computed property)

7.1.47 Events

7.1.48 Action

Plugin Version: 7.2, Platform: macOS, Targets: .

Function: The action event called when the user clicks on the statusitem.

Notes: This event is coming from the Cocoa event system. What you can do is a bit limited when using GUI functions from Xojo. To avoid some redraw errors, you may want to start a timer and let your Xojo code run a millisecond after the menu code has finished.

Depending on what you do, you can see the menu not redrawing properly (staying highlighted) and crashes if the Xojo code modifies some global Cocoa states.

7.1.49 validateMenuItem(menuItem as NSMenuItemMBS) as boolean

Plugin Version: 13.1, Platform: macOS, Targets: .

Function: Implemented to override the default action of enabling or disabling a specific menu item.

Notes: Return true to enable menuItem, false to disable it.

This is needed to modify the menuitem for a menu attached to a NSSearchFieldMBS.

MenuItem parameter is often the same as self, but for a searchfield menu, self is the template and menuItem parameter the actual menu item.

7.1.50 Constants

Constants

Constant	Value	Description
NSMixedState	-1	One of the constants for the state property. The corresponding feature is in effect somewhere.
NSOffState	0	One of the constants for the state property. The corresponding feature is in effect nowhere.
NSOnState	1	One of the constants for the state property. The corresponding feature is in effect everywhere.

7.2 class NSMenuItemMBS

7.2.1 class NSMenuItemMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: A class to represent a menu in the Cocoa world.

Notes: All methods in this class will catch exceptions from Cocoa and raise a `NSExceptionMBS` instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr4](#)
- [News from the MBS Xojo Plugins Version 23.5](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.5](#)
- [More Menubar options in macOS Sonoma](#)
- [MBS Xojo Plugins, version 23.5pr2](#)
- [MBS Xojo / Real Studio Plugins, version 15.5pr1](#)
- [MBS Xojo / Real Studio Plugins, version 15.2pr1](#)
- [Apply fonts to font PopupMenu](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr1](#)
- [MBS Xojo / Real Studio Plugins, version 14.0pr2](#)

Xojo Developer Magazine

- [5.6, page 33: Third Party Plugins: Statusitems, Getting an icon in the top right of the Mac OS X menubar by Christian Schmitz](#)
- [22.1, page 9: News](#)

7.2.2 Methods

7.2.3 addItem(m as NSMenuItemMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Adds a menu item to the end.

Notes: If the menuitem is nil, nothing happens.

The menuitem can only be in one menu.

The menu doesn't references to the RB classes behind, so it's up to you to keep those references if you want to get the events and avoid crashes.

The StatusItem example keeps an array with all those menu item objects it needs so RB will not destroy them.

7.2.4 cancelTracking

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Dismisses the menu and ends all menu tracking.

Notes: Available in Mac OS X v10.5 and later.

7.2.5 cancelTrackingWithoutAnimation

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Dismisses the menu and ends all menu tracking without displaying the associated animation.

Notes: Available in Mac OS X v10.6 and later.

7.2.6 Constructor(Handle as Integer)

Plugin Version: 15.2, Platform: macOS, Targets: Desktop only.

Function: Constructor for creating an instanced based on an existing handle.

Notes: Useful if you get a NSMenu reference from a declare.

The object is retained.

See also:

- 7.2.7 Constructor(title as string="")

780

7.2.7 Constructor(title as string="")

Plugin Version: 7.4, Platform: macOS, Targets: Desktop only.

Function: Constructor for a new nsmenu.

Example:

```
dim m as NSMenuMBS
m=new NSMenuMBS("Hello World")
msgbox m.title // shows "Hello World"
```

Notes: Title is optional.

See also:

- 7.2.6 Constructor(Handle as Integer)

7.2.8 helpMenu as NSMenuItem

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the help menu if one is registered.

7.2.9 indexOfItem(item as NSMenuItem) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index identifying the location of a specified menu item in the receiver.

Notes: Returns the integer index of the menu item or, if no such menu item is in the menu, -1.

7.2.10 indexOfItemWithSelector(selector as string) as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Searches menu items for one with the given selector.

Notes: Returns -1 if not found.

You can use this to find menu items by their selector. Which is often more save than by index or title.

7.2.11 indexOfItemWithSubmenu(item as NSMenuItem) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the menu item in the receiver with the given submenu.

Notes: item: A menu object that is a menu item of the receiver (that is, a submenu).

The integer index of the menu item or, if no such menu item is in the menu, -1.

7.2.12 indexOfItemWithTag(tag as Integer) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the first menu item in the receiver identified by a tag.

Notes: Returns the integer index of the menu item or, if no such menu item is in the menu, -1.

7.2.13 `indexOfItemWithTitle(title as string) as Integer`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the index of the first menu item in the receiver that has a specified title.

Notes: The integer index of the menu item or, if no such menu item is in the menu, -1.

7.2.14 `insertItem(m as NSMenuItemMBS, index as Integer)`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Inserts a menu item into the menu at a specific location.

Notes: If the menuitem is nil, nothing happens. A menuitem can only be part of one menu.

index: An integer index identifying the location of the menu item in the menu. Zero based.

7.2.15 `Item(index as Integer) as NSMenuItemMBS`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Returns the item with the given index.

Notes: Index is from 0 to numberOfItems-1.

Remember: The objects returned are not the same Xojo objects used with `addItem` or `insertItem`.

7.2.16 `itemWithSelector(selector as string) as NSMenuItemMBS`

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Searches menu items for one with the given selector.

Notes: You can use this to find menu items by their selector. Which is often more save than by index or title.

7.2.17 `mainMenu as NSMenuItemMBS`

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the main menu.

7.2.18 menuBarVisible as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the menu bar is visible.

Notes: Available in Mac OS X v10.2 and later.

Returns true if the menu bar is visible, otherwise false.

7.2.19 NSMenuDidAddItemNotification as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with NSNotificationObserverMBS class to get an event when such a notification is sent.

Posted after a menu item is added to the menu. The notification object is the instance of NSMenuMBS that just added the new menu item.

This is very useful to customize the menu in Xojo. The runtime rebuilds menu bar often, so you can catch it and edit menu.

The userInfo dictionary contains the following information:

NSMenuItemIndex: An integer index of the menu item that was added.

7.2.20 NSMenuDidBeginTrackingNotification as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Posted when menu tracking begins. The notification object is the main menu bar (NSApplicationMBS mainMenu) or the root menu of a popup button. This notification does not contain a userInfo dictionary.

Use this constant with NSNotificationObserverMBS class to get an event when such a notification is sent.

7.2.21 NSMenuDidChangeItemNotification as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with `NSNotificationObserverMBS` class to get an event when such a notification is sent.

Posted after a menu item in the menu changes appearance. Changes include enabling/disabling, changes in state, and changes to title. The notification object is the instance of `NSMenuMBS` with the menu item that changed. The `userInfo` dictionary contains the following information:

Key	Value
<code>NSMenuItemIndex</code>	An integer index of the menu item that changed.

7.2.22 `NSMenuDidEndTrackingNotification` as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with `NSNotificationObserverMBS` class to get an event when such a notification is sent.

Posted when menu tracking ends, even if no action is sent. The notification object is the main menu bar (`NSApplicationMBS mainMenu`) or the root menu of a popup button. This notification does not contain a `userInfo`

7.2.23 `NSMenuDidRemoveItemNotification` as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with `NSNotificationObserverMBS` class to get an event when such a notification is sent.

Posted after a menu item is removed from the menu. The notification object is the instance of `NSMenu` that just removed the menu item. The `userInfo` dictionary contains the following information:

`NSMenuItemIndex`: An integer index of the menu item that was removed. Note that this index may no longer be valid and in any event no longer points to the menu item that was removed.

7.2.24 NSMenuDidSendActionNotification as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with NSNotificationObserverMBS class to get an event when such a notification is sent.

Posted just after the application dispatches a menu item's action method to the menu item's target. The notification object is the instance of NSMenuMBS containing the chosen menu item. The userInfo dictionary contains the following information:

Key	Value
MenuItem	The menu item that was chosen.

7.2.25 NSMenuWillSendActionNotification as string

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: One of the notifications sent for menus.

Notes: Use this constant with NSNotificationObserverMBS class to get an event when such a notification is sent.

Posted just before the application dispatches a menu item's action method to the menu item's target. The notification object is the instance of NSMenuMBS containing the chosen menu item. The userInfo dictionary contains the following information:

Key	Value
MenuItem	The menu item that was chosen.

7.2.26 paletteMenuWithColors(colors() as NSColorMBS, itemTitles() as String, templateImage as NSImageMBS = nil) as NSMenuMBS

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: Creates a palette menu displaying user-selectable color tags using the provided template image, tinted using the specified array of colors.

Example:

```
dim m as NSMenuItemMBS = NSMenuItemMBS.MenuItem(EditMenu)
```

```

// gather data for menu
dim colors() as NSColorMBS
colors.Append NSColorMBS.whiteColor
colors.Append NSColorMBS.redColor
colors.Append NSColorMBS.blueColor

dim titles() as string = array("White", "Red", "Blue")

// now create palette and attach our handler
paletteMenu = NSMenuMBS.paletteMenuWithColors(Colors, titles)
AddHandler paletteMenu.PaletteSelectionChange, WeakAddressOf PaletteSelectionChange

// now make a dummy menu and add palette
dim n as new MyNSMenuItemMBS("Hello")
n.Enabled = true
n.submenu = paletteMenu
m.submenu.addItem n
self.HelloMenu = n

```

Notes: Optionally allows observing changes to the selection state in the compact menu. The block is invoked after the selection has been updated. Currently selected items can be retrieved from the ‘selectedItems’ property.

Returns a palette menu.

Requires macOS Sonoma or newer.

7.2.27 performActionForItemAtIndex(index as Integer)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Causes the application to send the action message of a specified menu item to its target.

Notes: If a target is not specified, the message is sent to the first responder. As a side effect, this method posts `NSMenuWillSendActionNotification` and `NSMenuDidSendActionNotification`.

In Mac OS X v10.6 and later the `performActionForItemAtIndex` no longer triggers menu validation. This is because validation is typically done during menu tracking or key equivalent matching, so the subsequent `performActionForItemAtIndex` validation was redundant. To trigger validation explicitly, use invoke the `update` method.

In Mac OS X v10.6 `performActionForItemAtIndex`, when called, now triggers highlighting in the menu bar. It also sends out appropriate accessibility notifications indicating the item was selected.

7.2.28 popUpContextMenu(menu as NSMenuItemMBS, theEvent as NSEventMBS, view as NSViewMBS, font as NSFontMBS = nil)

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Displays a contextual menu over a view for an event using a specified font.

Notes: menu: The menu object to use for the contextual menu.

event: An NSEvent object representing the event.

view: The view object over which to display the contextual menu.

font: An NSFont object representing the font for the contextual menu. If you pass in nil for the font, the method uses the default font for menu.

Specifying a font using the font parameter is discouraged. Instead set the menu's font using the setFont: method and pass nil for the font parameter.

7.2.29 popUpMenuPositioningItem(item as NSMenuItemMBS, location as NSPointMBS, view as NSViewMBS = nil) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop only.

Function: Pops up the menu at the specified location.

Example:

```
// create menu
dim m as new NSMenuItemMBS
m.autoenablesItems = false

// add some items
dim it as new NSMenuItemMBS
it.CreateMenuItem "Hello"
it.Enabled = true
m.addItem it

it = new NSMenuItemMBS
it.CreateMenuItem "World"
it.Enabled = true
m.addItem it

// show relative to this view
dim v as NSViewMBS = window1.PushButton1.NSViewMBS

dim r as Boolean
r = m.popUpMenuPositioningItem(nil, nil, v)
// you need to use NSMenuItemMBS subclasses with action event to get an event for which item was selected...
```

Notes: item: The menu item to be positioned at the specified location in the view.
 location: The location in the view coordinate system to display the menu item.
 view: The view to display the menu item over.

Returns true if menu tracking ended because an item was selected, and false if menu tracking was cancelled for any reason.

Pops up the receiver as a popup menu. The top left corner of the specified item (if specified, item must be present in the receiver) is positioned at the specified location in the specified view, interpreted in the view's own coordinate system.

If item is nil, the menu is positioned such that the top left of the menu content frame is at the given location. If view is nil, the location is interpreted in the screen coordinate system. This allows you to pop up a menu disconnected from any window.

Available in OS X v10.6 and later.

7.2.30 removeAllItems

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Removes all the menu items in the receiver.

Example:

```
dim n as NSMenuMBS // your menu

// for Mac OS X 10.6
n.removeAllItems

// for any Mac OS X version
for i as Integer = n.numberOfItems-1 DownTo 0
n.removeItemAtIndex i
next
```

Notes: This method is more efficient than removing menu items individually. Unlike the other remove methods, this method does not post NSMenuDidChangeItemNotification notifications.

Available in Mac OS X v10.6 and later.

7.2.31 removeItem(m as NSMenuItemMBS)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes the menuitem.

Notes: Does nothing if menuitem is nil.

The Xojo object used to add the menuitem must not be the same as the one you use here, but the values for the menuitems handle property must match.

7.2.32 removeItemAtIndex(index as Integer)

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: Removes the menu item with the given index.

Example:

```
dim n as NSMenuMBS // your menu

// for Mac OS X 10.6
n.removeAllItems

// for any Mac OS X version
for i as Integer = n.numberOfItems-1 DownTo 0
n.removeItemAtIndex i
next
```

Notes: Index is from 0 to NumberOfItems-1.

7.2.33 selectedItem as NSMenuItemMBS()

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: The menu items that are selected.

Notes: An item is selected when its state is 'NSControl.StateValue.on'.

Requires macOS Sonoma or newer.

7.2.34 `setMenuBarVisible(value as boolean)`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets whether the menu bar is visible and selectable by the user.

Notes: value: true if menu bar is to be visible, otherwise false.

Available in Mac OS X v10.2 and later.

7.2.35 `setSelectedItem(items() as NSMenuItemMBS)`

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: Sets the menu items that are selected.

Notes: An item is selected when its state is `'NSControl.StateValue.on'`.

Setting `'selectedItems'` will select any items that are contained in the provided array, and deselect any previously selected items that are not in the array.

Requires macOS Sonoma or newer.

7.2.36 `update`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Enables or disables the receiver's menu items based on the `NSMenuValidation` informal protocol and sizes the menu to fit its current menu items if necessary.

7.2.37 `windowsMenu as NSMenuMBS`

Plugin Version: 14.0, Platform: macOS, Targets: Desktop only.

Function: Returns the window menu if one is registered.

7.2.38 `Properties`

7.2.39 `allowsContextMenuPlugIns as boolean`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Whether the popup menu allows appending of contextual menu plugin items.

Notes: Value is true if the popup menu allows appending of contextual menu plugin items, otherwise false. Available in Mac OS X v10.6 and later.

Can be used to hide services menu items.
(Read and Write property)

7.2.40 autoenablesItems as Boolean

Plugin Version: 7.7, Platform: macOS, Targets: Desktop only.

Function: Whether menu items are automatically enabled and disabled.

Notes: Mac OS X 10.5 only.
(Read and Write property)

7.2.41 Font as NSFontMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The font used to display the menu and its submenus.

Notes: This font will be used to display the menu and any submenus that have not had their font set explicitly.

Available in Mac OS X v10.6 and later.
(Read and Write property)

7.2.42 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The internal reference to the NSMenu object.

Notes: (Read and Write property)

7.2.43 highlightedItem as NSMenuItemMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the highlighted item in the receiver.

Notes: Returns the highlighted item in the receiver, or nil if no item in the menu is highlighted. Available in Mac OS X v10.5 and later.
(Read only property)

7.2.44 Identifier as String

Plugin Version: 24.1, Platform: macOS, Targets: Desktop only.

Function: A string that identifies this user interface item.

Notes: It should be set to a unique value on `NSViews` when they are intended to be used inside a view-based `NSTableView`. Identifiers should be unique per-window. For programmatically created user interface items, you would typically set this value in code after creating a view but before adding it to a window. You may also want to set an identifier on a window, after creating it programmatically, to identify the window easily when it is reopened. You should not change the identifier after a view is added to a window. Identifiers beginning with an underscore are reserved for the system. In framework classes that implement this protocol, the accessor methods are not intended to be overridden.

To help avoid collision of identifiers, it is recommended that identifiers use the same prefix as is used for the framework or application. For example, identifiers for standard `AppKit` interface items, such as the open panel, will begin with "NS".

The slash '/', backslash '\', and colon ':' characters are reserved and should not be used in identifiers.
(Read and Write property)

7.2.45 menuBarHeight as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the menu bar height for the current application's main menu.

Notes: Returns the receiver's main menu bar height or 0.0 if the receiver is some other menu.

This method supersedes the `menuBarHeight` class method of the `NSMenuView` class.

Available in Mac OS X v10.4 and later.

(Read only property)

7.2.46 minimumWidth as Double

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: The minimum width of the menu.

Notes: The menu will not draw smaller than its minimum width, but may draw larger if it needs more space. The default value is 0.

Available in Mac OS X v10.6 and later.

(Read and Write property)

7.2.47 numberOfItems as Integer

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The number of menu items in this menu.

Notes: (Read only property)

7.2.48 presentationStyle as Integer

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: The presentation style of the menu.

Notes: This property is not respected if the menu is the main menu of the app.

Requires macOS Sonoma or newer.

(Read and Write property)

7.2.49 selectionMode as Integer

Plugin Version: 23.5, Platform: macOS, Targets: Desktop only.

Function: The selection mode of the menu.

Notes: Note the selection mode only has effect on menu items that belong to the same selection group. A selection group consists of the items with the same target/action.

Requires macOS Sonoma or newer.

(Read and Write property)

7.2.50 showsStateColumn as boolean

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver displays the state column.

Notes: Available in Mac OS X v10.5 and later.

Returns true if the receiver displays the state column, otherwise false.

The state column is the area in the menu items, where state of menuitems are shown like a checkmark.

(Read and Write property)

7.2.51 size as NSSizeMBS

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the size of the menu.

Notes: Returns the size of the menu in screen coordinates.

The menu may draw at a smaller size when shown, depending on its positioning and display configuration.

Available in Mac OS X v10.6 and later.

(Read only property)

7.2.52 supermenu as NSMenuMBS

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The parent menu of this menu.

Notes: (Read only property)

7.2.53 Title as String

Plugin Version: 7.2, Platform: macOS, Targets: Desktop only.

Function: The title of the menu.

Notes: (Read and Write property)

7.2.54 userInterfaceLayoutDirection as Integer

Plugin Version: 16.3, Platform: macOS, Targets: Desktop only.

Function: Configures the layout direction of menu items in the menu.

Notes: This property configures the layout direction (a value of type `NSUserInterfaceLayoutDirection`) of menu items in the menu. If no layout direction is explicitly set for a menu, then the menu defaults to the layout direction specified for the application object. See `userInterfaceLayoutDirection` in `NSApplication` Class Reference.

`NSUserInterfaceLayoutDirectionLeftToRight = 0`

`NSUserInterfaceLayoutDirectionRightToLeft = 1`

Available in OS X v10.11 and later.

(Read and Write property)

7.2.55 Events

7.2.56 DidClose

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Invoked after a menu closed.

Notes: Don't modify the structure of the menu or the menu items during this method.

7.2.57 EnableMenuItems

Plugin Version: 8.7, Platform: macOS, Targets: .

Function: This event is called before the menu opens so you can edit the menu.

7.2.58 PaletteSelectionChange

Plugin Version: 23.5, Platform: macOS, Targets: .

Function: The event called when the palette selection changes.

7.2.59 willHighlightItem(item as NSMenuItemMBS)

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Invoked to indicate that a menu is about to highlight a given item.

Notes: item: The item about to be highlighted.

Only one item per menu can be highlighted at a time. If item is nil, it means that all items in the menu are about to be unhighlighted.

7.2.60 WillOpen

Plugin Version: 14.3, Platform: macOS, Targets: .

Function: Invoked when a menu is about to open.

Notes: Don't modify the structure of the menu or the menu items during this method.

7.2.61 Constants

Presentation Styles

Constant	Value	Description
PresentationStylePalette	1	The menu marked as palette is to be displayed in place of the menu item presenting it, with its items aligned horizontally.
PresentationStyleRegular	0	The default presentation style. Typically means the menu will be presented as either a popup or pulldown menu, based on the context.

Selection Modes

Constant	Value	Description
SelectionModeAutomatic	0	The menu will determine the appropriate selection mode based on the context and its contents.
SelectionModeSelectAny	2	The user can select multiple items in the menu. A change in selection will not automatically deselect any previously selected item in the same selection group.
SelectionModeSelectOne	1	The user will be allowed to select at most one menu item in the same selection group at a time. A change in selection will deselect any previously selected item.

Chapter 8

Cocoa Networking

8.1 class NSCachedURLResponseMBS

8.1.1 class NSCachedURLResponseMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: A cached response to a URL request.

Notes: A NSCachedURLResponseMBS object provides the server,Ãs response metadata in the form of a NSURLResponseMBS object, along with an NSData object containing the actual cached content data. Its storage policy determines whether the response should be cached on disk, in memory, or not at all.

Cached responses also contain a user info dictionary where you can store app-specific information about the cached item.

The NSURLCacheMBS class stores and retrieves instances of NSCachedURLResponseMBS.

Blog Entries

- [MBS Xojo Plugins in version 19.0](#)
- [MBS Xojo Plugins, version 18.6pr2](#)

Xojo Developer Magazine

- [17.2, page 11: News](#)

8.1.2 Methods

8.1.3 Constructor(response as NSURLResponseMBS, data as MemoryBlock, userInfo as Dictionary = nil, storagePolicy as Integer = 0)

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Creates a cached URL response instance.

Notes: response: The response to cache.

data: The data to cache.

Returns a cached URL response object, containing the response and data.

The cache storage policy is set to the default, `kCacheStorageAllowed`, and the user info dictionary is set to `nil`.

8.1.4 copy as NSCachedURLResponseMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

8.1.5 Properties

8.1.6 Data as MemoryBlock

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The cached response,Ãs data.

Notes: (Read only property)

8.1.7 Handle as Integer

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

8.1.8 Response as NSURLResponseMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The URL response object associated with the instance.

Notes: (Read only property)

8.1.9 StoragePolicy as Integer

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The cached response,Ãs storage policy.

Notes: (Read only property)

8.1.10 UserInfo as Dictionary

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The cached response,Ãs user info dictionary.

Notes: This value is nil if there is no user info dictionary.
(Read only property)

8.1.11 Constants

Storage Policies

Constant	Value	Description
kCacheStorageAllowed	0	Allow caching.
kCacheStorageAllowedInMemoryOnly	1	Allow caching only in memory.
kCacheStorageNotAllowed	1	Caching is not allowed.

8.2 class NSURLAuthenticationChallengeMBS

8.2.1 class NSURLAuthenticationChallengeMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: A class for an authentication challenge.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.2.2 Methods

8.2.3 cancelAuthenticationChallenge

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Tells the system to cancel this challenge.

8.2.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

8.2.5 continueWithoutCredentialForAuthenticationChallenge

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Tells the system to continue this challenge without a password.

8.2.6 error as NSErrorMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the NSError object representing the last authentication failure.

Notes: This method returns nil if the protocol doesn't use errors to indicate an authentication failure.

Available in Mac OS X v10.2 with Safari 1.0 installed.

Available in Mac OS X v10.2.7 and later.

8.2.7 failureResponse as NSURLResponseMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the response representing authentication failure.

Notes: If there was a previous authentication failure, and this protocol uses responses to indicate authentication failure, then this method will return the response. Otherwise it will return nil.

8.2.8 previousFailureCount as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get count of previous failed authentication attempts.

8.2.9 proposedCredential as NSURLCredentialMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the proposed credential for this challenge.

Notes: proposedCredential may be nil, if there is no default credential to use for this challenge (either stored or in the URL). If the credential is not nil and returns true for hasPassword, this means the NSURLConnection thinks the credential is ready to use as-is. If it returns false for hasPassword, then the credential is not ready to use as-is, but provides a default username the client could use when prompting.

8.2.10 protectionSpace as NSURLProtectionSpaceMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get a description of the protection space that requires authentication.

Notes: Returns the protection space that needs authentication.

8.2.11 useCredential(credential as NSURLCredentialMBS)

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Tells the system to use a certain credential for this challenge.

8.2.12 Properties

8.2.13 Handle as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: The internal used handle for this challenge.

Notes: (Read and Write property)

8.3 class NSURLCacheMBS

8.3.1 class NSURLCacheMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The class for the caching.

Notes: NSURLCache implements the caching of responses to URL load requests by mapping NSURLRequest objects to NSCachedURLResponse objects. It is a composite of an in-memory and an on-disk cache.

Methods are provided to manipulate the sizes of each of these caches as well as to control the path on disk to use for persistent storage of cache data.

Blog Entries

- [MBS Xojo Plugins in version 19.0](#)
- [MBS Xojo Plugins, version 18.6pr2](#)

Videos

- [Presentation from Xojo Developer Conference 2019 in Miami.](#)

Xojo Developer Magazine

- [17.5, page 41: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes](#)
- [17.2, page 10: News](#)

8.3.2 Methods

8.3.3 cachedResponseForRequest(request as NSURLRequestMBS) as NSCachedURLResponseMBS

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Returns the cached URL response in the cache for the specified URL request.

Notes: request: The URL request whose cached response is desired.

Returns the cached URL response for request, or nil if no response has been cached.

8.3.4 Constructor(memoryCapacity as UInt64, diskCapacity as UInt64, diskPath as folderitem)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Initializes an `NSURLCache` object with the specified values.

Notes: `memoryCapacity`: The memory capacity of the cache, in bytes.

`diskCapacity`: The disk capacity of the cache, in bytes.

`path`: The location at which to store the on-disk cache.

The returned `NSURLCache` is backed by disk, so developers can be more liberal with space when choosing the capacity for this kind of cache. A disk cache measured in the tens of megabytes should be acceptable in most cases.

8.3.5 `removeAllCachedResponses`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Clears the receiver's cache, removing all stored cached URL responses.

8.3.6 `removeCachedResponseForRequest(request as NSURLRequestMBS)`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Removes the cached URL response for a specified URL request.

Notes: `request`: The URL request whose cached URL response should be removed. If there is no corresponding cached URL response, no action is taken.

8.3.7 `removeCachedResponsesSinceDate(d as date)`

Plugin Version: 19.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Clears the given cache of any cached responses since the provided date.

Notes: `d`: The earliest date of responses that should remain in the cache. Any responses with dates later than this parameter should be removed.

Available in macOS 10.10 or newer.

See also:

- 8.3.8 `removeCachedResponsesSinceDate(d as dateTime)`

804

8.3.8 `removeCachedResponsesSinceDate(d as dateTime)`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Clears the given cache of any cached responses since the provided date.

Notes: d: The earliest date of responses that should remain in the cache. Any responses with dates later than this parameter should be removed.

Available in macOS 10.10 or newer.

See also:

- 8.3.7 removeCachedResponsesSinceDate(d as date)

804

8.3.9 setSharedURLCache(cache as NSURLCacheMBS)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Sets the shared NSURLCache instance to a specified cache object.

Notes: Applications that have special caching requirements or constraints should use this method to specify an NSURLCache instance with customized cache settings.

8.3.10 sharedURLCache as NSURLCacheMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the shared NSURLCache instance.

Notes: The disk path is set to: <user_home_directory>/Library/Caches/<current_process_name>. The user's home directory is determined by calling NSHomeDirectory and the current process name is determined using NSProcessInfoMBS.processName.

Applications that do not have special caching requirements or constraints should find the default shared cache instance acceptable. Applications with more specific needs can create a custom NSURLCache object and set it as the shared cache instance using setSharedURLCache.

8.3.11 storeCachedResponse(cachedResponse as NSCachedURLResponseMBS, request as NSURLRequestMBS)

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: Stores a cached URL response for a specified request.

Notes: cachedResponse: The cached URL response to store.

request: The request for which the cached URL response is being stored.

If you override this method, you should also override storeCachedResponse.

8.3.12 Properties

8.3.13 `currentDiskUsage` as `UInt64`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the current size of the receiver's on-disk cache, in bytes.

Notes: (Read only property)

8.3.14 `currentMemoryUsage` as `UInt64`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the current size of the receiver's in-memory cache, in bytes.

Notes: (Read only property)

8.3.15 `diskCapacity` as `UInt64`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The on-disk cache capacity.

Notes: (Read and Write property)

8.3.16 `Handle` as `Integer`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal handle to the `NSURLCache` object.

Notes: (Read and Write property)

8.3.17 `memoryCapacity` as `UInt64`

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The in-memory cache capacity.

Notes: (Read and Write property)

8.3.18 Constants

Constants

Constant	Value	Description
NSURLCacheStorageAllowed	0	One of the constants for the cache strategy. Specifies that storage in an NSURLCache is allowed without restriction.
NSURLCacheStorageAllowedInMemoryOnly	1	One of the constants for the cache strategy. Specifies that storage in an NSURLCache is allowed; however storage should be done in memory only, no disk storage should be done.
NSURLCacheStorageNotAllowed	2	One of the constants for the cache strategy. Specifies that storage in an NSURLCache is not allowed in any fashion, in memory or on disk.

8.4 class `NSURLConnectionFilterMBS`

8.4.1 class `NSURLConnectionFilterMBS`

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class to filter URL connections.

Notes: This class is designed to allow you to intercept your application creating `NSURLConnection` objects and change the `NSURLRequest` used. This can be useful to change timeouts on `Xojo.Net.HTTPSocket` class.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.2](#)
- [MBS Xojo Plugins, version 18.2pr3](#)
- [Xojo.Net.HTTPSocket and Timeouts](#)

Videos

- [Presentation from Munich conference about MBS Plugins.](#)

8.4.2 Properties

8.4.3 Enabled as Boolean

Plugin Version: 18.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the event is enabled.

Notes: Default is true.

(Read and Write property)

8.4.4 Events

8.4.5 `FilterConnection(request as NSURLRequestMBS) as NSURLRequestMBS`

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

Function: The event called when a new connection is made.

Example:

```
Function FilterConnection(request as NSURLRequestMBS) Handles FilterConnection as NSURLRequestMBS
system.debuglog CurrentMethodName
system.debuglog "URL: "+request.URL
```

```
dim newRequest as NSMutableURLRequestMBS = request.mutableCopy
```

```
// change to 10 minutes  
newRequest.setTimeoutInterval 600  
  
return newRequest  
End Function
```

Notes: You get the current request and you can return a new request.
If you return nil, we pass through the existing one.
Only requests on main thread trigger the event.

8.5 class `NSURLConnectionMBS`

8.5.1 class `NSURLConnectionMBS`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Deprecated: This item is deprecated and should no longer be used. You can use `NSURLSessionMBS` instead. **Function:** An `NSURLConnection` object provides support to perform the loading of a URL request. **Notes:** The interface for `NSURLConnection` is sparse, providing only the controls to start and cancel asynchronous loads of a URL request.

`NSURLConnection`'s events allow an object to receive informational callbacks about the asynchronous load of a URL request. Other events provide facilities that allow the subclass to customize the process of performing an asynchronous URL load. These events are called on the thread that started the asynchronous load operation for the associated `NSURLConnection` object.

`NSURLConnection` also has a convenience class method, `sendSynchronousRequest`, to load a URL request synchronously.

The following contract governs the events defined in this class:

- Zero or more `willSendRequest` events are called before any further event is called if it is determined that the download must redirect to a new location. The delegate can allow the redirect, modify the destination, or deny the redirect.
- Zero or more `willSendRequestForAuthenticationChallenge` events are called before a request for an authentication challenge is sent. The delegate can call the appropriate `NSURLAuthenticationChallengeMBS` method and perform any other required task related to credentials.
- Zero or more `didReceiveAuthenticationChallenge` vents are called if it is necessary to authenticate in order to download the request and the connection does not already have authenticated credentials.
- Zero or more `didCancelAuthenticationChallenge` vents are called if the connection cancels the authentication challenge due to the protocol implementation encountering an error.
- Zero or more `didReceiveResponse` events are called before receiving a `didReceiveData` message. The only case where `didReceiveResponse` is not sent to a delegate is when the protocol implementation encounters an error before a response can be created.
- Zero or more `didReceiveData` events are called before any of the following messages are sent to the delegate: `willCacheResponse`, `DidFinishLoading`, `didFailWithError`.
- Zero or one `willCacheResponse` events are called to the delegate after `didReceiveData` is sent but before a `DidFinishLoading` message is sent.

Unless an `NSURLConnection` object receives a cancel message, the subclasses receives one and only one of `DidFinishLoading`, or `didFailWithError` message, but never both. In addition, once either of these messages

is sent, the delegate receives no further messages for the connection.

Blog Entries

- [Adding NSURLSession classes for Xojo](#)
- [MBS Xojo Plugins, version 20.2pr3](#)
- [MBS Real Studio Plugins, version 12.5pr3](#)
- [MBS Real Studio Plugins, version 11.3pr7](#)

8.5.2 Methods

8.5.3 cancel

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Cancels an asynchronous load of a request.

Notes: After this method is called, the connection's delegate no longer receives any messages for the connection. If you want to reattempt the connection, you should create a new connection object.

8.5.4 canHandleRequest(request as NSURLRequestMBS) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns whether a request can be handled based on a preflight evaluation.

Notes: request: The request to evaluate.

Returns true if a preflight operation determines that a connection with request can be created and the associated I/O can be started, false otherwise.

The result of this method is valid as long as no NSURLProtocol classes are registered or unregistered, and request remains unchanged. Applications should be prepared to handle failures even if they have performed request preflighting by calling this method.

8.5.5 Constructor(request as NSURLRequestMBS)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates URL connection and begins to load the data for the URL request.

Notes: request: The URL request to load. The request object is deep-copied as part of the initialization process. Changes made to request after this method returns do not affect the request that is used for the loading process.

On success handle property is not zero.

This is equivalent to calling Constructor and passing true for optional startImmediately.
See also:

- 8.5.6 Constructor(request as NSURLRequestMBS, startImmediately as boolean) 812

8.5.6 Constructor(request as NSURLRequestMBS, startImmediately as boolean)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates URL connection and begins to load the data for the URL request, if specified.

Notes: request: The URL request to load. The request object is deep-copied as part of the initialization process. Changes made to request after this method returns do not affect the request that is used for the loading process.

startImmediately: True if the connection should be loading data immediately, otherwise false.

On success the handle property is not zero.

Available in Mac OS X v10.5 and later.

See also:

- 8.5.5 Constructor(request as NSURLRequestMBS) 811

8.5.7 data as MemoryBlock

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: The data downloaded so far.

Notes: The plugin adds the new data it gets in the didReceiveData event to a big memoryblock and gives you access to it using this event.

8.5.8 sendSynchronousRequest(request as NSURLRequestMBS, byref response as NSURLResponseMBS, byref error as NSErrorMBS) as Memoryblock

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Performs a synchronous load of the specified URL request.

Example:

```
dim request as new NSURLRequestMBS("http://www.monkeybreadsoftware.de/images/MBSLogo.jpg")
dim error as NSErrorMBS
dim response as NSURLResponseMBS
```

```

dim d as MemoryBlock = NSURLConnectionMBS.sendSynchronousRequest(request, response, error)

if d<>Nil then
dim pic as Picture = JPEGStringToPictureMBS(d, true)
window1.Backdrop = pic
end if

if error<>Nil then
MsgBox "Error: "+error.description
end if

```

Notes: request: The URL request to load. The request object is deep-copied as part of the initialization process. Changes made to request after this method returns do not affect the request that is used for the loading process.

response: Out parameter for the URL response returned by the server.

error: Out parameter used if an error occurs while processing the request.

Returns the downloaded data for the URL request. Returns nil if a connection could not be created or if the download fails.

A synchronous load is built on top of the asynchronous loading code made available by the class. The calling thread is blocked while the asynchronous loading system performs the URL load on a thread spawned specifically for this load request. No special threading or run loop configuration is necessary in the calling thread in order to perform a synchronous load.

Important: Because this call can potentially take several minutes to fail (particularly when using a cellular network in iOS), you should never call this function from the main thread of a GUI application.

If authentication is required in order to download the request, the required credentials must be specified as part of the URL. If authentication fails, or credentials are missing, the connection will attempt to continue without credentials.

8.5.9 start

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Causes the connection to begin loading data, if it has not already.

Notes: Calling this method is necessary only if you create a connection with the Constructor method and provide false for the startImmediately parameter.

Available in Mac OS X v10.5 and later.

8.5.10 Properties

8.5.11 Handle as Integer

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

8.5.12 Events

8.5.13 `canAuthenticateAgainstProtectionSpace(ProtectionSpace as NSURLProtectionSpaceMBS)` as boolean

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent to determine whether the delegate is able to respond to a protection space's form of authentication.

Notes: `ProtectionSpace`: The protection space that generates an authentication challenge.

Return true if you are able to respond to a protection space's form of authentication, otherwise false.

This method is called before `didReceiveAuthenticationChallenge`, allowing the class to inspect a protection space before attempting to authenticate against it. By returning true, the event indicates that it can handle the form of authentication, which it does in the subsequent call to `didReceiveAuthenticationChallenge`. If the event returns false, the system attempts to use the user's keychain to authenticate. If your delegate does not implement this method and the protection space uses client certificate authentication or server trust authentication, the system behaves as if you returned false. The system behaves as if you returned true for all other authentication methods.

Available in Mac OS X v10.6 and later.

8.5.14 `didCancelAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a connection cancels an authentication challenge.

Notes: `challenge`: The challenge that was canceled.

8.5.15 didFailWithError(error as NSErrorMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a connection fails to load its request successfully.

Notes: error: An error object containing details of why the connection failed to load the request successfully.

Once the delegate receives this message, it will receive no further messages for connection.

8.5.16 didFinishLoading

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a connection has finished loading successfully.

8.5.17 didReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a connection must authenticate a challenge in order to download its request.

Notes: challenge: The challenge that connection must authenticate in order to download its request.

This method gives the class the opportunity to determine the course of action taken for the challenge provide credentials, continue without providing credentials, or cancel the authentication challenge and the download.

The delegate can determine the number of previous authentication challenges by sending the message `previousFailureCount` to `challenge`.

If the previous failure count is 0 and the value returned by `proposedCredential` is `nil`, the delegate can create a new `NSURLCredential` object, providing information specific to the type of credential, and send a `useCredential:forAuthenticationChallenge:` message to [challenge sender], passing the credential and challenge as parameters. If `proposedCredential` is not `nil`, the value is a credential from the URL or the shared credential storage that can be provided to the user as feedback.

The delegate may decide to abandon further attempts at authentication at any time by sending [challenge sender] a `continueWithoutCredentialForAuthenticationChallenge:` or a `cancelAuthenticationChallenge:` message. The specific action is implementation dependent.

If the delegate implements this method, the download will suspend until [challenge sender] is sent one of the following messages: `useCredential:forAuthenticationChallenge:`, `continueWithoutCredentialForAuthen-`

enticationChallenge: or cancelAuthenticationChallenge:.

If the delegate does not implement this method the default implementation is used. If a valid credential for the request is provided as part of the URL, or is available from the NSURLCredentialStorage the [challenge sender] is sent a useCredential:forAuthenticationChallenge: with the credential. If the challenge has no credential or the credentials fail to authorize access, then continueWithoutCredentialForAuthenticationChallenge: is sent to [challenge sender] instead.

8.5.18 didReceiveData(newData as Memoryblock)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent as a connection loads data incrementally.

Notes: newData: The newly available data.

The data property is updated before this event is called and gives you a way to see all data received so far.

8.5.19 didReceiveResponse(response as NSURLResponseMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when the connection has received sufficient data to construct the URL response for its request.

Notes: response: The URL response for the connection's request. This object is immutable and will not be modified by the URL loading system once it is presented to the delegate.

In rare cases, for example in the case of an HTTP load where the content type of the load data is multi-part/x-mixed-replace, the delegate will receive more than one didReceiveResponse message. In the event this occurs, delegates should discard all data previously delivered by didReceiveData, and should be prepared to handle the, potentially different, MIME type reported by the newly reported URL response.

The only case where this message is not sent to the delegate is when the protocol implementation encounters an error before a response could be created.

8.5.20 didSendBodyData(bytesWritten as Int64, totalBytesWritten as Int64, totalBytesExpectedToWrite as Int64)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent as the body (message data) of a request is transmitted (such as in an http POST request).

Notes: bytesWritten: The number of bytes written in the latest write.
totalBytesWritten: The total number of bytes written for this connection.
totalBytesExpectedToWrite: The number of bytes the connection expects to write.

This method provides an estimate of the progress of a URL upload.
The value of totalBytesExpectedToWrite may change during the upload if the request needs to be retransmitted due to a lost connection or an authentication challenge from the server.
Available in Mac OS X v10.6 and later.

8.5.21 shouldUseCredentialStorage as boolean

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent to determine whether the URL loader should consult the credential storage for authenticating the connection.

Notes: This method is called before any attempt to authenticate is made. By returning false, the delegate tells the connection not to consult the credential storage and makes itself responsible for providing credentials for any authentication challenges. Not implementing this method is the same as returning true. The delegate is free to consult the credential storage itself when it receives a didReceiveAuthenticationChallenge event.

Available in Mac OS X v10.6 and later.

8.5.22 willCacheResponse(cachedResponse as NSCachedURLResponseMBS) as NSCachedURLResponseMBS

Plugin Version: 19.0, Platform: macOS, Targets: .

Function: The connection will cache this response.

Notes: If you implement this event, please return request to cache or nil to not cache.

8.5.23 willSendRequest(request as NSURLRequestMBS, redirectResponse as NSURLResponseMBS) as NSURLRequestMBS

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when the connection determines that it must change URLs in order to continue loading a request.

Notes: request: The proposed redirected request. You should inspect the redirected request to verify that it meets its needs, and create a copy with new attributes to return to the connection if necessary.

redirectResponse: The URL response that caused the redirect. May be nil in cases where this method is not

being sent as a result of involving the delegate in redirect processing.

The actual URL request to use in light of the redirection response. The event may return request unmodified to allow the redirect, return a new request, or return nil to reject the redirect and continue processing the connection.

If you wish to cancel the redirect connection, it should call the connection object's cancel method. Alternatively, the delegate method can return nil to cancel the redirect connection, and the original connection will continue to process. This has special relevance in the case where redirectResponse is not nil. In this case, any data that is loaded for the connection will be sent to the delegate, and the delegate will receive a connectionDidFinishLoading or didFailLoadingWithError message, as appropriate.

Note: Prior to Mac OS X version 10.5, returning nil in this method sometimes would cancel the connection but other times would cause the connection to use the given request unmodified. In addition, prior to version 10.5, NSURLConnection would often modify the incoming NSURLRequest object before transmission without notifying the delegate. In version 10.5 and later, it always notifies the delegate via willSendRequest, and thus the delegate might receive this message before the connection has even properly begun, prior to transmitting the request to the remote server.

The delegate can receive this message as a result of modifying a request before it is sent, for example to transform the request's URL to its canonical form. To detect this case, examine redirectResponse; if it is nil, the message was not sent due to a redirect.

The delegate should be prepared to receive this message multiple times.

8.5.24 willSendRequestForAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Tells the delegate that the connection will send a request for an authentication challenge.

Notes: challenge: The authentication challenge for which a request is being sent.

This method allows the delegate to make an informed decision about connection authentication at once. If the delegate implements this method, it has no need to implement canAuthenticateAgainstProtectionSpace, didReceiveAuthenticationChallenge, shouldUseCredentialStorage. In fact, these other methods are not invoked.

In this method, you must invoke one of the challenge-responder methods (NSURLAuthenticationChallengeSenderMBS):

```
useCredential
continueWithoutCredentialForAuthenticationChallenge
```

```
cancelAuthenticationChallenge  
performDefaultHandlingForAuthenticationChallenge  
rejectProtectionSpaceAndContinueWithChallenge
```

You might also want to analyze challenge for the authentication scheme and the proposed credential before calling a `NSURLAuthenticationChallengeSenderMBS` method. You should never assume that a proposed credential is present. You can either create your own credential and respond with that, or you can send the proposed credential back. (Because this object is immutable, if you want to change it you must copy it and then modify the copy.)

8.6 class NSURLCredentialMBS

8.6.1 class NSURLCredentialMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: A class for a stored password.

Notes: dim u as NSURLCredentialMBS

dim p as Integer = NSURLCredentialMBS.NSURLCredentialPersistenceForSession

u = NSURLCredentialMBS.credential("Christian", "teddy123", p)

MsgBox u.user+EndOfLine+u.password+EndOfLine+str(u.persistence)

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 20.2pr3](#)
- [MBS Xojo / Real Studio Plugins, version 13.4pr6](#)

8.6.2 Methods

8.6.3 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

8.6.4 copy as NSURLCredentialMBS

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Creates a copy of the credential object.

8.6.5 credential(User as string, password as string, persistence as Integer = 0) as NSURLCredentialMBS

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Create a new NSURLCredential with a user and password.

Example:

dim u as NSURLCredentialMBS

dim p as Integer = NSURLCredentialMBS.NSURLCredentialPersistenceForSession

```
u = NSURLCredentialMBS.credential("Christian", "teddy123", p)
MsgBox u.user+EndOfLine+u.password+EndOfLine+str(u.persistence)
```

Notes: user: the username

password: the password

persistence: Integer that says to store per session, permanently or not at all.

Can be NSURLCredentialPersistenceForSession, NSURLCredentialPersistenceNone or NSURLCredentialPersistencePermanent.

8.6.6 credentialWithPEM(Data as MemoryBlock, Password as String = "") as NSURLCredentialMBS

Plugin Version: 20.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates an identity credentials with certificate and private key in PEM file.

Notes: Password is optional.

This is a convenience function using SecItemImport to read data and then builds the result with credentialWithIdentity function.

Returns nil for any error.

8.6.7 credentialWithPKCS12(Data as MemoryBlock, Password as String = "") as NSURLCredentialMBS

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Creates an identity credentials with certificate and private key in PKCS#12 file.

Notes: Password is optional.

This is a convenience function using SecPKCS12Import to read data and then builds the result with credentialWithIdentity function.

Returns nil for any error.

8.6.8 credentialWithTrustingServer(ProtectionSpace as NSURLProtectionSpaceMBS) as NSURLCredentialMBS

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Creates a credential trusting the server in the protection space.

Notes: This is a convenience function using the ServerTrust in the protection space to wrap it in a new to read data and then builds the result with credential object with credentialForTrust function.

Returns nil for any error.

8.6.9 Properties

8.6.10 CertificateCount as Integer

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Queries number of certificates in this credential object.

Notes: (Read only property)

8.6.11 Handle as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: The internal used handle for this class.

Notes: (Read and Write property)

8.6.12 HasIdentity as Boolean

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Whether this credential object contains an identity reference.

Notes: (Read only property)

8.6.13 hasPassword as Boolean

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Find out if this credential has a password, without trying to get it.

Notes: Returns true if this credential has a password, otherwise false.

If this credential's password is actually kept in an external store, the password method may return "" even if this method returns true, since getting the password may fail, or the user may refuse access.

(Read only property)

8.6.14 password as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the password.

Notes: This method might actually attempt to retrieve the password from an external store, possible resulting in prompting, so do not call it unless needed.

(Read only property)

8.6.15 persistence as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Determine whether this credential is or should be stored persistently.

Notes: Use the constants:

NSURLCredentialPersistenceNone

NSURLCredentialPersistenceForSession

NSURLCredentialPersistencePermanent

(Read only property)

8.6.16 user as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the username.

Notes: (Read only property)

8.6.17 Constants

Constants

Constant	Value	Description
NSURLCredentialPersistenceForSession	1	This credential will only be stored for this session.
NSURLCredentialPersistenceNone	0	This credential won't be saved.
NSURLCredentialPersistencePermanent	2	This credential will be stored permanently and shared with other applica-

8.7 class NSURLCredentialStorageMBS

8.7.1 class NSURLCredentialStorageMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: NSURLCredentialStorage implements a singleton object (shared instance) which manages the shared credentials cache.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

8.7.2 Methods

8.7.3 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

8.7.4 defaultCredentialForProtectionSpace(space as NSURLProtectionSpaceMBS) as NSURLCredentialMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the default credential for the specified protection space.

8.7.5 sharedCredentialStorage as NSURLCredentialStorageMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the shared singleton authentication storage.

8.7.6 Properties

8.7.7 Handle as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: The internal used handle for this class.

Notes: (Read and Write property)

8.8 class NSURLDownloadMBS

8.8.1 class NSURLDownloadMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSURL Download class which handles downloads for Webkit.

Notes: NSURLDownload downloads a request asynchronously and saves the data to a file. The interface for NSURLDownload is sparse, providing methods to initialize a download, set the destination path and cancel loading the request.

NSURLDownload’s delegate methods—defined by the NSURLDownloadDelegate—allow an object to receive informational callbacks about the asynchronous load of the URL request. Other delegate methods provide facilities that allow the delegate to customize the process of performing an asynchronous URL load.

Note that these delegate methods are called on the thread that started the asynchronous load operation for the associated NSURLDownload object.

The MBS Plugins currently only implement a part of this class for use with WebDownloadDelegateMBS. So please contact us if you need more.

Blog Entries

- [MBS Xojo Plugins, version 19.1pr2](#)
- [Notes from the last days](#)
- [MBS Real Studio Plugins, version 11.3pr6](#)
- [MBS Real Studio Plugins, version 11.3pr5](#)

8.8.2 Methods

8.8.3 cancel

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Cancels the receiver’s download and deletes the downloaded file.

8.8.4 canResumeDownloadDecodedWithEncodingMIMETYPE(MimeType as string) as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns whether a URL download object can resume a download that was decoded with the specified MIME type.

Notes: MIMEType: The MIME type the caller wants to know about.

Returns true if the URL download object can resume a download that was decoded with the specified MIME type, false otherwise.

NSURLDownload cannot resume a download that was partially decoded in the gzip format.

8.8.5 Constructor(request as NSURLRequestMBS)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an initialized URL download for a URL request and begins to download the data for the request.

Notes: request: The URL request to download. The request object is deep-copied as part of the initialization process. Changes made to request after this method returns do not affect the request that is used for the loading process.

On success the handle property is not zero.

See also:

- 8.8.6 Constructor(resumeData as Memoryblock, path as folderitem) 827
- 8.8.7 Constructor(resumeData as Memoryblock, path as string) 827

8.8.6 Constructor(resumeData as Memoryblock, path as folderitem)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an initialized NSURLDownload object that will resume downloading the specified data to the specified file and begins the download.

Notes: resumeData: Specifies the data to resume downloading.

path: The location for the downloaded data.

On success the handle property is not zero.

See also:

- 8.8.5 Constructor(request as NSURLRequestMBS) 827
- 8.8.7 Constructor(resumeData as Memoryblock, path as string) 827

8.8.7 Constructor(resumeData as Memoryblock, path as string)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an initialized NSURLDownload object that will resume downloading the specified data to the specified file and begins the download.

Notes: resumeData: Specifies the data to resume downloading.

path: The location for the downloaded data.

On success the handle property is not zero.

See also:

- 8.8.5 Constructor(request as NSURLRequestMBS) 827
- 8.8.6 Constructor(resumeData as Memoryblock, path as folderitem) 827

8.8.8 request as NSURLRequestMBS

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the request that initiated the receiver's download.

8.8.9 resumeData as Memoryblock

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the resume data for a download that is not yet complete.

Notes: The resume data for a download that is not yet complete. This data represents the necessary state information that an NSURLDownload object needs to resume a download. The resume data can later be used when initializing a download with Constructor. Returns nil if the download is not able to be resumed.

Resume data will only be returned if the protocol of the download as well as the server support resuming. In order to later resume a download you must call setDeletesFileUponFailure passing false so the partially downloaded data is not deleted when the initial connection is lost or canceled.

8.8.10 setDestination(path as folderitem, allowOverwrite as boolean)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the destination path of the downloaded file.

Notes: path: The path for the downloaded file.

allowOverwrite: true if an existing file at path can be replaced, false otherwise.

If allowOverwrite is false and a file already exists at path, a unique filename will be created for the downloaded file by appending a number to the filename. The delegate can implement didCreateDestination to determine the filename used when the file is written to disk.

8.8. CLASS NSURLDOWNLOADMBS

829

An NSURLDownload instance ignores multiple calls to this method.

See also:

- 8.8.11 setDestination(path as string, allowOverwrite as boolean)

829

8.8.11 setDestination(path as string, allowOverwrite as boolean)

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the destination path of the downloaded file.

Notes: path: The path for the downloaded file.

allowOverwrite: true if an existing file at path can be replaced, false otherwise.

If allowOverwrite is false and a file already exists at path, a unique filename will be created for the downloaded file by appending a number to the filename. The delegate can implement didCreateDestination to determine the filename used when the file is written to disk.

An NSURLDownload instance ignores multiple calls to this method.

See also:

- 8.8.10 setDestination(path as folderitem, allowOverwrite as boolean)

828

8.8.12 Properties

8.8.13 Handle as Integer

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

8.8.14 deletesFileUponFailure as boolean

Plugin Version: 11.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the receiver deletes partially downloaded files when a download stops prematurely.

Notes: True if partially downloaded files should be deleted when a download stops prematurely, false otherwise. The default is true.

(Read and Write computed property)

8.8.15 Events

8.8.16 `canAuthenticateAgainstProtectionSpace(protectedSpace as NSURLProtectionSpaceMBS) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent to determine whether the delegate is able to respond to a protection space's form of authentication. (required)

Notes: `protectedSpace`: The protection space that generates an authentication challenge.

This method is called before `didReceiveAuthenticationChallenge`, allowing the delegate to inspect a protection space before attempting to authenticate against it. By returning true, the delegate indicates that it can handle the form of authentication, which it does in the subsequent call to `didReceiveAuthenticationChallenge`. Not implementing this method is the same as returning false, in which case default authentication handling is used.

Available in Mac OS X v10.6 and later.

8.8.17 `decideDestinationWithSuggestedFilename(filename as string)`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: The delegate receives this message when download has determined a suggested filename for the downloaded file. (required)

Notes: `filename`: The suggested filename for the download.

The suggested filename is either derived from the last path component of the URL and the MIME type or, if the download was encoded, from the encoding. If the delegate wishes to modify the path, it should send `setDestination` to download.

The delegate will not receive this message if `setDestination` has already been called for the download.

Available in Mac OS X v10.2 and later.

8.8.18 `DidBegin`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent immediately after a download object begins a download. (required)

Notes: Available in Mac OS X v10.2 and later.

8.8.19 didCancelAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent if an authentication challenge is canceled due to the protocol implementation encountering an error. (required)

Notes: challenge: The authentication challenge that caused the download object to cancel the download.

If the delegate receives this message the download will fail and the delegate will receive a didFailWithError message.

Available in Mac OS X v10.2 and later.

8.8.20 didCreateDestination(path as string, file as folderitem)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when the destination file is created. (required)

Notes: path: The path to the destination file.

file: The path to the destination file as folderitem.

Available in Mac OS X v10.2 and later.

8.8.21 didFailWithError(error as NSErrorMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent if the download fails or if an I/O error occurs when the file is written to disk. (required)

Notes: error: The error that caused the failure of the download.

Any partially downloaded file will be deleted.

Once the delegate receives this message, it will receive no further messages for download.

Available in Mac OS X v10.2 and later.

8.8.22 DidFinish

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a download object has completed downloading successfully and has written its results to disk. (required)

Notes: The delegate will receive no further messages for download.
Available in Mac OS X v10.2 and later.

8.8.23 didReceiveAuthenticationChallenge(challenge as NSURLAuthenticationChallengeMBS)

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when the URL download must authenticate a challenge in order to download the request. (required)

Notes: challenge: The URL authentication challenge that must be authenticated in order to download the request.

This method gives the delegate the opportunity to determine the course of action taken for the challenge: provide credentials, continue without providing credentials or cancel the authentication challenge and the download.

The delegate can determine the number of previous authentication challenges by sending the message `previousFailureCount` to `challenge`.

If the previous failure count is 0 and the value returned by `proposedCredential` is `nil`, the delegate can create a new `NSURLCredential` object, providing information specific to the type of credential, and send a `useCredential` message to `challenge`, passing the credential and `challenge` as parameters. If `proposedCredential` is not `nil`, the value is a credential from the URL or the shared credential storage that can be provided to the user as feedback.

The delegate may decide to abandon further attempts at authentication at any time by sending `challenge` a `continueWithoutCredentialForAuthenticationChallenge` or a `cancelAuthenticationChallenge` message. The specific action is implementation dependent.

If the delegate implements this method, the download will suspend until [`challenge sender`] is sent one of the following messages: `useCredential`, `continueWithoutCredentialForAuthenticationChallenge` or `cancelAuthenticationChallenge`.

If the delegate does not implement this method the default implementation is used. If a valid credential for the request is provided as part of the URL, or is available from the `NSURLCredentialStorage` the challenge is sent a `useCredential:forAuthenticationChallenge` with the credential. If the challenge has no credential or

the credentials fail to authorize access, then `continueWithoutCredentialForAuthenticationChallenge` is sent to challenge sender instead.

Available in Mac OS X v10.2 and later.

8.8.24 `didReceiveDataOfLength(length as UInt64)`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent as a download object receives data incrementally. (required)

Notes: `length`: The amount of data received in this increment of the download, measured in bytes.

Available in Mac OS X v10.2 and later.

8.8.25 `didReceiveResponse(response as NSURLResponseMBS)`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a download object has received sufficient load data to construct the `NSURLResponse` object for the download. (required)

Notes: `response`: The URL response object received as part of the download. `response` is immutable and will not be modified after this method is called.

In some rare cases, multiple responses may be received for a single download. In this case, the client should assume that each new response resets the download progress to 0 and should check the new response for the expected content length.

Available in Mac OS X v10.2 and later.

8.8.26 `shouldDecodeSourceDataOfMIMEType(encodingType as string) as boolean`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a download object determines that the downloaded file is encoded to inquire whether the file should be automatically decoded. (required)

Notes: `encodingType`: The type of encoding used by the downloaded file. The supported encoding formats are MacBinary ("application/macbinary"), Binhex ("application/mac-binhex40") and gzip ("application/gzip").

Return true to decode the file, false otherwise.

The delegate may receive this message more than once if the file has been encoded multiple times. This method is not called if the downloaded file is not encoded.
Available in Mac OS X v10.2 and later.

8.8.27 `shouldUseCredentialStorage` as `boolean`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent to determine whether the URL loader should consult the credential storage to authenticate the download. (required)

Notes: This method is called before any attempt to authenticate is made. By returning false, the delegate tells the download not to consult the credential storage and makes itself responsible for providing credentials for any authentication challenges. Not implementing this method is the same as returning true. The delegate is free to consult the credential storage itself when it receives a `didReceiveAuthenticationChallenge` message.

Available in Mac OS X v10.6 and later.

8.8.28 `willResumeWithResponse(response as NSURLResponseMBS, starting-Byte as Int64)`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when a download object has received a response from the server after attempting to resume a download. (required)

Notes: `response`: The URL response received from the server in response to an attempt to resume a download.

The location of the start of the resumed data, in bytes.

Available in Mac OS X v10.4 and later.

8.8.29 `willSendRequest(request as NSURLRequestMBS, redirectResponse as NSURLResponseMBS) as NSURLRequestMBS`

Plugin Version: 11.3, Platform: macOS, Targets: .

Function: Sent when the download object determines that it must change URLs in order to continue loading a request. (required)

Notes: `request`: The proposed redirected request. The delegate should inspect the redirected request to verify that it meets its needs, and create a copy with new attributes to return to the connection if necessary.
`redirectResponse`: The URL response that caused the redirect. May be nil in cases where this method is not being sent as a result of involving the delegate in redirect processing.

Return the actual URL request to use in light of the redirection response. The delegate may copy and modify request as necessary to change its attributes, return request unmodified, or return nil.

If the delegate wishes to cancel the redirect, it should call the download object's cancel method. Alternatively, the delegate method can return nil to cancel the redirect, and the download will continue to process. This has special relevance in the case where redirectResponse is not nil. In this case, any data that is loaded for the download will be sent to the delegate, and the delegate will receive a downloadDidFinish: or download:didFailWithError: message, as appropriate.

The delegate can receive this message as a result of transforming a request's URL to its canonical form, or for protocol-specific reasons, such as an HTTP redirect. The delegate implementation should be prepared to receive this message multiple times.

Available in Mac OS X v10.2 and later.

8.9 class NSURLMBS

8.9.1 class NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The class for URL.

Notes: An NSURL object represents a URL that can potentially contain the location of a resource on a remote server, the path of a local file on disk, or even an arbitrary piece of encoded data.

Please also review Apples documentation on this class:

https://developer.apple.com/library/mac/documentation/Cocoa/Reference/Foundation/Classes/NSURL_Class/index.html

MBS Plugin only includes a part of the original class. EMail us if you miss something.

Blog Entries

- [MBS Xojo Plugins, version 22.6pr1](#)
- [News from the MBS Xojo Plugins Version 22.1](#)
- [Tip of the day: Count files in upload for iCloud](#)
- [MBS Xojo Plugins, version 22.1pr3](#)
- [MBS Xojo Plugins, version 21.2pr3](#)
- [MBS Xojo Plugins, version 20.5pr6](#)
- [MBS Xojo Plugins, version 20.3pr6](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr4](#)
- [MBS Xojo / Real Studio Plugins, version 15.2pr1](#)

8.9.2 Methods

8.9.3 checkResourceIsReachableAndReturnError as NSErrorMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns whether the URL's resource exists and is reachable.

Notes: This method synchronously checks if the resource's backing store is reachable. Checking reachability is appropriate when making decisions that do not require other immediate operations on the resource, e.g. periodic maintenance of UI state that depends on the existence of a specific document. When performing operations such as opening a file or copying resource properties, it is more efficient to simply try the operation and handle failures. If this method returns false, the optional error is populated. This method is currently applicable only to URLs for file system resources. For other URL types, NO is returned. Symbol is present

in iOS 4, but performs no operation.

8.9.4 Constructor(item as folderitem)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates an URL based on folderItem.

See also:

- 8.9.5 Constructor(scheme as string, host as string, path as string) 837
- 8.9.6 Constructor(url as string) 837
- 8.9.7 Constructor(url as string, baseURL as NSURLMBS) 838

8.9.5 Constructor(scheme as string, host as string, path as string)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Initializes a newly created NSURL with a specified scheme, host, and path.

Notes: scheme: The scheme for the NSURL object. For example, in the URL `http://www.example.com/index.html`, the scheme is `http`.

host: The host for the NSURL object (for example, `www.example.com`). May be the empty string.

path: The path for the NSURL object (for example, `/index.html`). If the path begins with a tilde, you must first expand it by calling `stringByExpandingTildeInPath`.

See also:

- 8.9.4 Constructor(item as folderitem) 837
- 8.9.6 Constructor(url as string) 837
- 8.9.7 Constructor(url as string, baseURL as NSURLMBS) 838

8.9.6 Constructor(url as string)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates and returns an NSURL object initialized with a provided URL string.

See also:

- 8.9.4 Constructor(item as folderitem) 837
- 8.9.5 Constructor(scheme as string, host as string, path as string) 837
- 8.9.7 Constructor(url as string, baseURL as NSURLMBS) 838

8.9.7 Constructor(url as string, baseURL as NSURLMBS)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates and returns an NSURL object initialized with a base URL and a relative string.

Notes: This method allows you to create a URL relative to a base path or URL. For example, if you have the URL for a folder on disk and the name of a file within that folder, you can construct a URL for the file by providing the folder, the URL as the base path (with a trailing slash) and the filename as the string part.

This method expects NSString to contain only characters that are allowed in a properly formed URL. All other characters must be properly percent escaped. Any percent-escaped characters are interpreted using UTF-8 encoding.

See also:

- 8.9.4 Constructor(item as folderitem) 837
- 8.9.5 Constructor(scheme as string, host as string, path as string) 837
- 8.9.6 Constructor(url as string) 837

8.9.8 copy as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates a copy of the URL object.

8.9.9 fileURLWithFileSystemRepresentation(path as string, isDirectory as boolean, relativeToURL as NSURLMBS) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Initializes a newly created URL referencing the local file or directory at the file system representation of the path.

Notes: File system representation is a string with canonical UTF-8 encoding.

8.9.10 fileURLWithPath(path as string) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Builds a file URL with given file path.

Notes: Better to use fileURLWithPath with directory parameter if you know if the path is a directory vs non-directory, as it saves an i/o.

See also:

- 8.9.11 fileURLWithPath(path as string, isDirectory as boolean) as NSURLMBS 839

8.9.11 fileURLWithPath(path as string, isDirectory as boolean) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Builds a file URL based on a given file path.

See also:

- 8.9.10 fileURLWithPath(path as string) as NSURLMBS

838

8.9.12 fileURLWithPathComponents(components() as string) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Builds an URL based on given components.

8.9.13 getResourceValue(byref value as Variant, key as string, byref error as NSErrorMBS) as boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns the value of the resource property for the specified key.

Notes: value: The location where the value for the resource property identified by key should be stored.

key: The name of one of the URL's resource properties.

error: The error that occurred if the resource value could not be retrieved.

Returns true if value is successfully populated; otherwise, false.

This method first checks if the URL object already caches the resource value. If so, it returns the cached resource value to the caller. If not, then this method synchronously obtains the resource value from the backing store, adds the resource value to the URL object's cache, and returns the resource value to the caller.

The type of the returned resource value varies by resource property; for details, see the documentation for the key you want to access.

If this method returns true and the value is populated with nil, it means that the resource property is not available for the specified resource, and that no errors occurred when determining that the resource property was unavailable.

If this method returns false, an error occurred. The object pointer referenced by error is populated with additional information.

This method applies only to URLs that represent file system resources.

Available in OS X v10.6 and later.

Automatic type translation applies (See FAQ about NSDictionary).
e.g. NSNumber can be converted to Integer or Boolean depending on content.

8.9.14 isEqual(other as NSURLMBS) as boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Tests if two NSURLs are equal.

Notes: Returns a Boolean value that indicates whether the receiver and a given object have identical URL strings and base URLs.

8.9.15 Items(byref error as NSErrorMBS, VisibleItemsOnly as boolean = false) as NSURLMBS()

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Queries list of items in the directory the URL points to on disk.

Notes: Error is optional.

See also:

- 8.9.16 Items(VisibleItemsOnly as boolean = false) as NSURLMBS() 840

8.9.16 Items(VisibleItemsOnly as boolean = false) as NSURLMBS()

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Queries list of items in the directory the URL points to on disk.

See also:

- 8.9.15 Items(byref error as NSErrorMBS, VisibleItemsOnly as boolean = false) as NSURLMBS() 840

8.9.17 mountedVolumeURLs(SkipHidden as boolean = true) as NSURLMBS()

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Queries array of all NSURLs for mounted volumes.

Notes: if SkipHidden is true, hidden volumes are skipped.

Returns nil in case of error.

8.9.18 NSThumbnail1024x1024SizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible keys for the NSURLThumbnailDictionaryKey dictionary.

Notes: A 1024 x 1024 pixel thumbnail as an NSImage on OS X.
Available in OS X v10.10 and later.

8.9.19 NSURLAddedToDirectoryDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The time at which the resource,Ã was created or renamed into or within its parent directory, returned as a date. Inconsistent behavior may be observed when this attribute is requested on hard-linked items. This property is not supported by all volumes. (read-only)

Available in OS X v10.10.

8.9.20 NSURLAttributeModificationDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The time at which the resource,Ã attributes were most recently modified, returned as an NSDate object if the volume supports attribute modification dates, or nil if attribute modification dates are unsupported (read-write).

Available in OS X v10.6 and later.

8.9.21 NSURLContentAccessDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The time at which the resource was most recently accessed, returned as a date object if the volume supports access dates, or nil if access dates are unsupported (read-only).

Available in OS X v10.6 and later.

8.9.22 NSURLContentModificationDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The time at which the resource was most recently modified, returned as a date object if the volume supports modification dates, or nil if modification dates are unsupported (read-write).

Available in OS X v10.6 and later.

8.9.23 NSURLCreationDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource's creation date, returned as an NSDate object if the volume supports creation dates, or nil if creation dates are unsupported (read-write).

Available in OS X v10.6 and later.

8.9.24 NSURLCustomIconKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The icon stored with the resource, returned as an NSImage object, or nil if the resource has no custom icon.

Available in OS X v10.6 and later.

8.9.25 NSURLDocumentIdentifierKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The document identifier returned as a number (read-only).

The document identifier is a value assigned by the kernel to a file or directory. This value is used to identify the document regardless of where it is moved on a volume. The identifier persists across system restarts. It is not transferred when the file is copied, but it survives "safe save" operations. For example, it remains on the path to which it was assigned, even after calling the `replaceItemAtURL:withItemAtURL:backupItemName:options:resultingItemURL:error:` method. Document identifiers are only unique within a single vol-

ume. This property is not supported by all volumes.

Available in OS X v10.10 and iOS 8.0.

8.9.26 NSURLEffectiveIconKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ãs normal icon, returned as an NSImage object (read-only).

Available in OS X v10.6 and later.

8.9.27 NSURLFileAllocatedSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that apply to properties of files.

Notes: Key for the total size allocated on disk for the file, returned as an NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.28 NSURLFileResourceIdentifierKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ãs unique identifier, returned as an id (read-only).

This identifier can be used to determine equality between file system resources with the isEqual: method. Two resources are equal if they have the same file-system path or if their paths link to the same inode on the same file system.

The value of this identifier is not persistent across system restarts.

Available in OS X v10.7 and later.

If used with getResourceValue will give a MemoryBlock.

8.9.29 NSURLFileResourceTypeBlockSpecial as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a block special file.

Available in OS X v10.7 and later.

8.9.30 NSURLFileResourceTypeCharacterSpecial as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a character special file.

Available in OS X v10.7 and later.

8.9.31 NSURLFileResourceTypeDirectory as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a directory.

Available in OS X v10.7 and later.

8.9.32 NSURLFileResourceTypeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ãs object type, returned as an NSString object. See File Resource Types for possible values.

Available in OS X v10.7 and later.

8.9.33 NSURLFileResourceTypeNamedPipe as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a named pipe.

Available in OS X v10.7 and later.

8.9.34 NSURLFileResourceTypeRegular as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a regular file.

Available in OS X v10.7 and later.

8.9.35 NSURLFileResourceTypeSocket as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a socket.

Available in OS X v10.7 and later.

8.9.36 NSURLFileResourceTypeSymbolicLink as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource is a symbolic link.

Available in OS X v10.7 and later.

8.9.37 NSURLFileResourceTypeUnknown as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the possible values for the NSURLFileResourceTypeKey key.

Notes: The resource's type is unknown.

Available in OS X v10.7 and later.

8.9.38 NSURLFileScheme as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: These schemes are the ones that NSURL can parse.

Notes: Identifies a URL that points to a file on a mounted volume.

8.9.39 NSURLFileSecurityKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource's security information, returned as an NSFileSecurity object (read-write). Available in OS X v10.7 and later.

8.9.40 NSURLFileSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that apply to properties of files.

Notes: Key for the file's size in bytes, returned as an NSNumber object (read-only). Available in OS X v10.6 and later.

8.9.41 NSURLGenerationIdentifierKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: An opaque generation identifier, returned as an id (read-only)

The generation identifier can be compared using `isEqual` to determine if the data in a document has been modified. For URLs which refer to the same file inode, the generation identifier changes when the data in the file's data fork is changed. Changes to extended attributes or other file system metadata do not change the identifier. For URLs which refer to the same directory inode, the generation identifier changes when direct children of that directory are added, removed or renamed. Changes to the data of the direct children of that directory does not change the generation identifier. The identifier persists across system restarts. It is tied to a specific document on a specific volume and is not transferred when the document is copied to another volume. This property is not supported by all volumes.

Available in OS X v10.10 and iOS 8.0.

8.9.42 NSURLHasHiddenExtensionKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource's extension is normally removed from its localized name, returned as a Boolean (read-write).

8.9.43 NSURLIsAliasFileKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that apply to properties of files.

Notes: Key for determining whether the file is an alias, returned as a Boolean NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.44 NSURLIsDirectoryKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is a directory, returned as a Boolean- (read-only).

Available in OS X v10.6 and later.

8.9.45 NSURLIsExcludedFromBackupKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is excluded from all backups of app data, returned as a Boolean NSNumber object (read-write).

You can use this property to exclude cache and other app support files which are not needed in a backup. Some operations commonly made to user documents cause this property to be reset to false; consequently, do not use this property on user documents.

Available in OS X v10.8 and later.

8.9.46 NSURLIsExecutableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the current process (as determined by the EUID) can execute the resource (if it is a file) or search the resource (if it is a directory), returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.47 NSURLIsHiddenKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is normally not displayed to users, returned as a Boolean NSNumber object (read-write).

If the resource is hidden because its name begins with a period, setting this value has no effect.
Available in OS X v10.6 and later.

8.9.48 NSURLIsMountTriggerKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the URL is a file system trigger directory, returned as a Boolean NSNumber object (read-only). Traversing or opening a file system trigger directory causes an attempt to mount a file system on the directory.

Available in OS X v10.7 and later.

8.9.49 NSURLIsPackageKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is a file package, returned as a Boolean NSNumber object (read-write in OS X v10.8 and later, read-only in previous versions). A true value means that the resource is a file package.

If you attempt to set or clear this key,Ãs value on a file instead of a directory, the system ignores your attempt. If the directory is defined as a package by way of its filename extension or other reason apart from this key, setting this key,Ãs value to false has no effect.

Available in OS X v10.6 and later.

8.9.50 NSURLIsReadableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the current process (as determined by the EUID) can read the resource, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.51 NSURLIsRegularFileKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is a regular file, as opposed to a directory or a symbolic link. Returned as a Boolean NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.52 NSURLIsSymbolicLinkKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is a symbolic link, returned as a Boolean NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.53 NSURLIsSystemImmutableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource's system immutable bit is set, returned as a Boolean NSNumber object (read-write).

Available in OS X v10.6 and later.

8.9.54 NSURLIsUbiquitousItemKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the keys for files.

Example:

```
Dim f as FolderItem // your folderitem
Dim n As New NSURLMBS(f)

dim value as Variant
dim error as NSErrorMBS
Dim result As Boolean = n.getResourceValue(value, n.NSURLIsUbiquitousItemKey, error)
break // check in debugger
```

Notes: A boolean that contains true if this item is in iCloud storage, false if it is a local item (read-only).

Available in OS X v10.7 and later.

8.9.55 NSURLIsUserImmutableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource's user immutable bit is set, returned as a Boolean NSNumber object (read-write).

Available in OS X v10.6 and later.

8.9.56 NSURLIsVolumeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the resource is the root directory of a volume, returned as a Boolean NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.57 NSURLIsWritableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: Key for determining whether the current process (as determined by the EUID) can write to the

resource, returned as a Boolean NSNumber object (read-only).

Available in OS X v10.7 and later.

8.9.58 NSURLKeysOfUnsetValueKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Key for the resource properties that have not been set after setResourceValues returns an error, returned as an array of strings.

8.9.59 NSURLLabelColorKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource's label color, returned as an NSColor object, or nil if the resource has no label color (read-only).

Available in OS X v10.6 and later.

8.9.60 NSURLLabelNumberKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource's label number, returned as an NSNumber object (read-write).

Available in OS X v10.6 and later.

8.9.61 NSURLLinkCountKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The number of hard links to the resource, returned as an NSNumber object (read-only).

Available in OS X v10.6 and later.

8.9.62 NSURLLocalizedLabelKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ås localized label text, returned as an NSString object, or nil if the resource has no localized label text (read-only).

Available in OS X v10.6 and later.

8.9.63 NSURLLocalizedNameKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ås localized or extension-hidden name, returned as an NSString object (read-only).

Available in OS X v10.6 and later.

8.9.64 NSURLLocalizedTypeDescriptionKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ås localized type description, returned as an NSString object (read-only).

Available in OS X v10.6 and later.

8.9.65 NSURLNameKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource,Ås name in the file system, returned as an NSString object (read-write).

Available in OS X v10.6 and later.

8.9.66 NSURLParentDirectoryURLKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The parent directory of the resource, returned as an NSURL object, or nil if the resource is the root directory of its volume (read-only).
Available in OS X v10.6 and later.

8.9.67 NSURLPathKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The file system path for the URL, returned as an NSString object (read-only).

Available in OS X v10.8 and later.

8.9.68 NSURLPreferredIOBlockSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The optimal block size to use when reading or writing this file's data, returned as an NSNumber object, or nil if the preferred size is not available (read-only).

Available in OS X v10.7 and later.

8.9.69 NSURLQuarantinePropertiesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A key for quarantine properties.

Example:

```
Dim f As FolderItem = SpecialFolder.UserHome.Child("Downloads").Child("Installation.pdf")
```

```
Dim n As New NSURLMBS(f)
```

```
'Dim d As Dictionary = n.QuarantineProperties
```

```
Dim d As Dictionary
```

```
Dim v As Variant
```

```
dim e as NSErrorMBS
```

```
If n.getResourceValue(v, n.NSURLQuarantinePropertiesKey, e) Then
```

```
  d = v
```

```
  Break // inspect in debugger
```

```
Else
```

```
Break // failed  
End If
```

Notes: The quarantine properties as defined in LSQuarantine.h. To remove quarantine information from a file, pass NSNull as the value when setting this property. (Read-write, value type dictionary)
Available on Mac OS X 10.10 and later.

8.9.70 NSURLTagNamesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The names of tags attached to the resource, returned as an array of String values (read-write).

Available in OS X v10.9 and later.

8.9.71 NSURLThumbnailDictionaryKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: A dictionary of NSImage objects keyed by size (read-write).

See Thumbnail Property Keys for a list of possible keys.

Available in OS X v10.10.

8.9.72 NSURLThumbnailKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: All thumbnails as a single NSImage (read-write).

Available in OS X v10.10.

8.9.73 NSURLTotalFileAllocatedSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that apply to properties of files.

Notes: Key for the total allocated size of the file in bytes, returned as an NSNumber object (read-only).

This includes the size of any file metadata.

Available in OS X v10.7 and later.

8.9.74 NSURLTotFileSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that apply to properties of files.

Notes: Key for the total displayable size of the file in bytes, returned as an NSNumber object (read-only). This includes the size of any file metadata.

Available in OS X v10.7 and later.

8.9.75 NSURLTypeIdentifierKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The resource's uniform type identifier (UTI), returned as a string (read-only). Available in OS X v10.6 and later.

8.9.76 NSURLUbiquitousItemContainerDisplayNameKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A string containing the name of the item's container, as it is displayed to the user.

Available in OS X v10.10 and later.

8.9.77 NSURLUbiquitousItemDownloadingErrorKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: An error object that indicates why downloading the item from iCloud failed.

Available in OS X v10.9 and later.

8.9.78 `NSURLUbiquitousItemDownloadingStatusCurrent` as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Values that describe the iCloud storage state of a file.

Notes: A local copy of this item exists and is the most up-to-date version known to the device.

Available in OS X v10.9 and later.

8.9.79 `NSURLUbiquitousItemDownloadingStatusDownloaded` as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Values that describe the iCloud storage state of a file.

Notes: A local copy of this item exists, but it is stale. The most recent version will be downloaded as soon as possible.

Available in OS X v10.9 and later.

8.9.80 `NSURLUbiquitousItemDownloadingStatusKey` as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: The current download state for the item, indicating whether a local copy exists and whether that copy is the most current version of the item. The possible values for this key are described in Ubiquitous Item Downloading Status Constants.

Available in OS X v10.9 and later.

8.9.81 `NSURLUbiquitousItemDownloadingStatusNotDownloaded` as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Values that describe the iCloud storage state of a file.

Notes: This item has not been downloaded yet. Use `startDownloadingUbiquitousItemAtURL` to download it.

Available in OS X v10.9 and later.

8.9.82 NSURLUbiquitousItemDownloadRequestedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A Boolean indicating whether a call to `startDownloadingUbiquitousItemAtURL` has already been made to download the item. The value of this key is read-only.

Available in OS X v10.10 and later.

8.9.83 NSURLUbiquitousItemHasUnresolvedConflictsKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A boolean NSNumber that contains true if this item has conflicts outstanding, false otherwise (read-only).

Available in OS X v10.7 and later.

8.9.84 NSURLUbiquitousItemIsDownloadedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A boolean NSNumber that contains true if this item's data has been downloaded to a ubiquity container, false otherwise (read-only).

Available in OS X v10.7 and later.

Deprecated in OS X v10.9.

8.9.85 NSURLUbiquitousItemIsDownloadingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A boolean NSNumber that contains true if this item is being downloaded from iCloud, false otherwise (read-only).

Available in OS X v10.7 and later.

8.9.86 NSURLUbiquitousItemIsExcludedFromSyncKey as string

Plugin Version: 21.5, Platform: macOS, Targets: All.

Function: One of the resource keys.

Notes: Value is a boolean.

The item is excluded from sync, which means it is locally on disk but won't be available on the server. An excluded item is no longer ubiquitous.

8.9.87 NSURLUbiquitousItemIsSharedKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the ubiquitous item is shared. (Read-only, value type boolean)

8.9.88 NSURLUbiquitousItemIsUploadedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A boolean NSNumber that contains true if this item's data has been uploaded to iCloud storage, false otherwise (read-only).

When waiting for an upload to complete, do not poll this key from within a block passed to `coordinateReadingItemAtURL`, because the coordinated read required to obtain this value cannot be performed until that block completes and returns. Instead, use `NSMetadataQuery` or an `NSFilePresenter` delegate to asynchronously notify your app when the status changes.

Available in OS X v10.7 and later.

8.9.89 NSURLUbiquitousItemIsUploadingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: A boolean NSNumber that contains true if this item is being uploaded to iCloud, false otherwise

(read-only).

Available in OS X v10.7 and later.

8.9.90 NSURLUbiquitousItemPercentDownloadedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: An NSNumber in the range 0–100 that indicates the percentage of the data that has been downloaded (read-only).

Use the NSMetadataQuery class to search for NSMetadataItem objects that have the NSMetadataUbiquitousItemPercentDownloadedKey attribute instead.

Available in OS X v10.7 and later.

Deprecated in OS X v10.8.

8.9.91 NSURLUbiquitousItemPercentUploadedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: An NSNumber in the range 0–100 that indicates the percentage of the data that has been uploaded (read-only).

Use the NSMetadataQuery class to search for NSMetadataItem objects that have the NSMetadataUbiquitousItemPercentUploadedKey attribute instead.

Available in OS X v10.7 and later.

Deprecated in OS X v10.8.

8.9.92 NSURLUbiquitousItemUploadingErrorKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Keys that describe the iCloud storage state of a file.

Notes: An error object that indicates why uploading the item to iCloud failed.

Available in OS X v10.9 and later.

8.9.93 NSURLUbiquitousSharedItemCurrentUserPermissionsKey as string

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

Function: One of the file system URL resource keys.

Notes: returns the permissions for the current user, or nil if not shared. (Read-only, value type string). Possible values below.

8.9.94 NSURLUbiquitousSharedItemCurrentUserRoleKey as string

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

Function: One of the file system URL resource keys.

Notes: Returns the current user's role for this shared item, or nil if not shared. (Read-only, value type string). See NSURLUbiquitousSharedItemRoleOwner and NSURLUbiquitousSharedItemRoleParticipant.

8.9.95 NSURLUbiquitousSharedItemMostRecentEditorNameComponentsKey as string

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

Function: One of the file system URL resource keys.

Notes: returns a NSPersonNameComponents for the most recent editor of the document, or nil if it is the current user. (Read-only, value type NSPersonNameComponents)

8.9.96 NSURLUbiquitousSharedItemOwnerNameComponentsKey as string

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

Function: One of the file system URL resource keys.

Notes: returns a NSPersonNameComponents, or nil if the current user. (Read-only, value type NSPersonNameComponents)

8.9.97 NSURLUbiquitousSharedItemPermissionsReadOnly as string

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

Function: One of the file system URL resource keys.

Notes: the current user is only allowed to read this item

8.9.98 NSURLUbiquitousSharedItemPermissionsReadWrite as string

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

Function: One of the file system URL resource keys.

Notes: the current user is allowed to both read and write this item

8.9.99 NSURLUbiquitousSharedItemRoleOwner as string

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

Function: One of the file system URL resource keys.

Notes: the current user is the owner of this shared item.

8.9.100 NSURLUbiquitousSharedItemRoleParticipant as string

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

Function: One of the file system URL resource keys.

Notes: the current user is a participant of this shared item.

8.9.101 NSURLVolumeAvailableCapacityForImportantUsageKey as string

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

Function: One of the file system URL resource keys.

Notes: Total available capacity in bytes for "Opportunistic" resources, including space expected to be cleared by purging non-essential and cached resources. "Opportunistic" means something that the user is likely to want but does not expect to be present on the local system, but is ultimately non-essential and replaceable. This would include items that will be created or downloaded without an explicit request from the user on the current device.

Examples: A background download of a newly available episode of a TV series that a user has been recently watching, a piece of content explicitly requested on another device, or a new document saved to a network server by the current user from another device.

Value is zero if unknown. Please switch to NSURLVolumeAvailableCapacityKey key in that case.

Read-only, value type number

8.9.102 NSURLVolumeAvailableCapacityForOpportunisticUsageKey as string

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

Function: One of the file system URL resource keys.

Notes: Total available capacity in bytes for "Important" resources, including space expected to be cleared by purging non-essential and cached resources. "Important" means something that the user or application clearly expects to be present on the local system, but is ultimately replaceable. This would include items that the user has explicitly requested via the UI, and resources that an application requires in order to provide functionality.

Examples: A video that the user has explicitly requested to watch but has not yet finished watching or an audio file that the user has requested to download.

This value should not be used in determining if there is room for an irreplaceable resource. In the case of irreplaceable resources, always attempt to save the resource regardless of available capacity and handle failure as gracefully as possible.

Value is zero if unknown. Please switch to NSURLVolumeAvailableCapacityKey key in that case.

Read-only, value type number

8.9.103 NSURLVolumeAvailableCapacityKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the volume,Ãs available capacity in bytes, returned as an Int64 (read-only). Available in OS X v10.6 and later.

8.9.104 NSURLVolumeCreationDateKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the volume,Ãs creation date, returned as a date, or nil if it cannot be determined (read-only).

Available in OS X v10.7 and later.

8.9.105 NSURLVolumeIdentifierKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The unique identifier of the resource,Ãs volume, returned as an id (read-only).

This identifier can be used with the isEqual method to determine whether two file system resources are on the same volume.

The value of this identifier is not persistent across system restarts.

Available in OS X v10.7 and later.

If used with getResourceValue will give a MemoryBlock.

8.9.106 NSURLVolumeIsAutomountedKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is automounted, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.107 NSURLVolumeIsBrowsableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is visible in GUI-based file-browsing environments, such as the Desktop or the Finder application, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.108 NSURLVolumeIsEjectableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is ejectable from the drive mechanism under software control, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.109 NSURLVolumeIsEncryptedKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume is encrypted. (Read-only, value type boolean)

8.9.110 NSURLVolumeIsInternalKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is connected to an internal bus, returned as a Boolean, or nil if it cannot be determined (read-only).

Available in OS X v10.7 and later.

8.9.111 NSURLVolumeIsJournalingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is currently journaling, returned as a Boolean. Available in OS X v10.6 and later.

8.9.112 NSURLVolumeIsLocalKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Example:

```
dim f as FolderItem = GetFolderItem("/Volumes/Ablage1", FolderItem.PathTypeNative)
```

```
dim n as NSURLMBS = NSURLMBS.URLWithItem(f)
```

```
dim value as Variant
```

```
dim error as NSErrorMBS
```

```
dim Local as Boolean
```

```
// network volumes are not local
```

```
if n.getResourceValue(value, n.NSURLVolumeIsLocalKey, error) then
```

```
MsgBox "Local: "+value.StringValue
```

```
Local = value.BooleanValue
```

```
else
```

```
MsgBox error.LocalizedDescription
```

```
end if
```

```
// network volumes have an URL
```

```
if n.getResourceValue(value, n.NSURLVolumeURLForRemountingKey, error) then
```

```
MsgBox "URLForRemounting: "+value.StringValue
```

```
else
```

```
MsgBox error.LocalizedDescription
```

```
end if
```

Notes: Key for determining whether the volume is stored on a local device, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.113 NSURLVolumeIsReadOnlyKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume is read-only, returned as a Boolean (read-only). Available in OS X v10.7 and later.

8.9.114 NSURLVolumeIsRemovableKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Example:

```
dim f as FolderItem = GetFolderItem("/Volumes/Test", FolderItem.PathTypeNative)

dim n as NSURLMBS = NSURLMBS.URLWithItem(f)
dim value as Variant
dim error as NSErrorMBS

if n.getResourceValue(value, n.NSURLVolumeIsRemovableKey, error) then
  MsgBox "Removable: "+value.StringValue
else
  MsgBox error.LocalizedDescription
end if
```

Notes: Key for determining whether the volume is removable from the drive mechanism, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.115 NSURLVolumeIsRootFileSystemKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume is the root filesystem. (Read-only, value type boolean)

8.9.116 NSURLVolumeLocalizedFormatDescriptionKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the volume's descriptive format name, returned as a string (read-only). Available in OS X v10.6 and later.

8.9.117 NSURLVolumeLocalizedNameKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: The name of the volume as it should be displayed in the user interface, returned as a string (read-only).

Available in OS X v10.7 and later.

8.9.118 NSURLVolumeMaximumFileSizeKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the largest file size supported by the volume in bytes, returned as a Boolean, or nil if it cannot be determined (read-only).

Available in OS X v10.7 and later.

8.9.119 NSURLVolumeNameKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: The name of the volume, returned as an NSString object (read-write). Settable only if NSURLVolumeSupportsRenamingKey is true.

Available in OS X v10.7 and later.

8.9.120 NSURLVolumeResourceCountKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the total number of resources on the volume, returned as an integer (read-only). Available in OS X v10.6 and later.

8.9.121 NSURLVolumeSupportsAccessPermissionsKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports setting POSIX access permissions with the NSURLFileSecurityKey property (Read-only, value type boolean)

8.9.122 NSURLVolumeSupportsAdvisoryFileLockingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume implements whole-file advisory locks in the style of flock, along with the O_EXLOCK and O_SHLOCK flags of the open function, returned as a Boolean NSNumber object (read-only).

Available in OS X v10.7 and later.

8.9.123 NSURLVolumeSupportsCasePreservedNamesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Example:

```
dim f as FolderItem = Volume(0)
dim n as new NSURLMBS(f)

dim v as Variant
dim e as NSErrorMBS
if n.getResourceValue(v, n.NSURLVolumeSupportsCasePreservedNamesKey, e) then
MsgBox "VolumeSupportsCasePreservedNames: "+str(v.BooleaValue)
else
MsgBox "Failed to query: "+e.LocalizedDescription
end if
```

Notes: Key for determining whether the volume supports case-preserved names, returned as a Boolean (read-only).

Available in OS X v10.6 and later.

8.9.124 NSURLVolumeSupportsCaseSensitiveNamesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Example:

```
dim f as FolderItem = Volume(0)
dim n as new NSURLMBS(f)

dim v as Variant
dim e as NSErrorMBS
if n.getResourceValue(v, n.NSURLVolumeSupportsCaseSensitiveNamesKey, e) then
  MsgBox "VolumeSupportsCaseSensitiveNames: "+str(v.BooleanValue)
else
  MsgBox "Failed to query: "+e.LocalizedDescription
end if
```

Notes: Key for determining whether the volume supports case-sensitive names, returned as a Boolean (read-only).

Available in OS X v10.6 and later.

8.9.125 NSURLVolumeSupportsCompressionKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports transparent decompression of compressed files using decmpfs. (Read-only, value type boolean)

8.9.126 NSURLVolumeSupportsExclusiveRenamingKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports `renamex_np(2)`'s `RENAME_EXCL` option (Read-only, value type boolean)

8.9.127 NSURLVolumeSupportsExtendedSecurityKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports extended security (access control lists), returned as a Boolean (read-only) (read-only).

Available in OS X v10.7 and later.

8.9.128 NSURLVolumeSupportsFileCloningKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports clonefile(2) (Read-only, value type boolean)

8.9.129 NSURLVolumeSupportsFileProtectionKey as string

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

Function: One of the file system URL resource keys.

Notes: True if the volume supports the File Protection attribute (see NSURLFileProtectionKey). (Read-only, value type number)

8.9.130 NSURLVolumeSupportsHardLinksKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports hard links, returned as a Boolean.

Available in OS X v10.6 and later.

8.9.131 NSURLVolumeSupportsImmutableFilesKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports making files immutable with the NSURLIsUserImmutableKey or NSURLIsSystemImmutableKey properties (Read-only, value type boolean)

8.9.132 NSURLVolumeSupportsJournalingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports journaling, returned as a Boolean. Available in OS X v10.6 and later.

8.9.133 NSURLVolumeSupportsPersistentIDsKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports persistent IDs, returned as a Boolean (read-only). Available in OS X v10.6 and later.

8.9.134 NSURLVolumeSupportsRenamingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume can be renamed, returned as a Boolean (read-only). Available in OS X v10.7 and later.

8.9.135 NSURLVolumeSupportsRootDirectoryDatesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports reliable storage of times for the root directory, returned as a Boolean (read-only).

Available in OS X v10.7 and later.

8.9.136 NSURLVolumeSupportsSparseFilesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports sparse files, returned as a Boolean.

Available in OS X v10.6 and later.

8.9.137 NSURLVolumeSupportsSwapRenamingKey as string

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

Function: One of the file system URL resource keys.

Notes: true if the volume supports `renamex_np(2)`'s `RENAME_SWAP` option (Read-only, value type boolean)

8.9.138 NSURLVolumeSupportsSymbolicLinksKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports symbolic links, returned as a Boolean. Available in OS X v10.6 and later.

8.9.139 NSURLVolumeSupportsVolumeSizesKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports returning volume size information, returned as a Boolean (read-only). If true, volume size information is available as values of the `NSURLVolumeTotalCapacityKey` and `NSURLVolumeAvailableCapacityKey` keys.

Available in OS X v10.7 and later.

8.9.140 NSURLVolumeSupportsZeroRunsKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for determining whether the volume supports zero runs, returned as a Boolean. (read-only). Available in OS X v10.6 and later.

8.9.141 NSURLVolumeTotalCapacityKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the volume's capacity in bytes, returned as an Int64 (read-only).

Available in OS X v10.6 and later.

8.9.142 NSURLVolumeURLForRemountingKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the URL needed to remount the network volume, returned as an NSURL object, or nil if a URL is not available (read-only).

Available in OS X v10.7 and later.

8.9.143 NSURLVolumeURLKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the file system URL resource keys.

Notes: The root directory of the resource's volume, returned as an NSURL object (read-only).

Available in OS X v10.6 and later.

8.9.144 NSURLVolumeUUIDStringKey as string

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: One of the property keys for volumes.

Notes: Key for the volume's persistent UUID, returned as an NSString object, or nil if a persistent UUID is not available (read-only).

Available in OS X v10.7 and later.

8.9.145 pathComponents as string()

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: An array containing the path components. (read-only)

Notes: This property contains an array containing the individual path components of the URL. For example, in the URL file:///directory/directory2/file, the path components array would be "/", "directory", "directory2", "file".

8.9.146 removeAllCachedResourceValues

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Removes all cached resource values and temporary resource values from the URL object.

Notes: This method is applicable only to URLs that represent file system resources.

Available in OS X v10.9 and later.

8.9.147 removeCachedResourceValueForKey(key as string)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Removes the cached resource value identified by a given key from the URL object.

Notes: key: The resource value key whose cached values you want to remove.

Removing a cached resource value may remove other cached resource values because some resource values are cached as a set of values, and because some resource values depend on other resource values. (Temporary resource values have no dependencies.)

This method is currently applicable only to URLs for file system resources.

Available in OS X v10.9 and later.

8.9.148 resourceValuesForKeys(keys() as string, byref error as NSErrorMBS) as Dictionary

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns the resource values for the properties identified by specified array of keys.

Notes: Returns a dictionary of resource values indexed by key.

This method first checks if the URL object already caches the specified resource values. If so, it returns the

cached resource values to the caller. If not, then this method synchronously obtains the resource values from the backing store, adds the resource values to the URL object's cache, and returns the resource values to the caller.

The type of the returned resource value varies by resource property; for details, see the documentation for the key you want to access.

If the result dictionary does not contain a resource value for one or more of the requested resource keys, it means those resource properties are not available for the URL, and no errors occurred when determining those resource properties were not available.

If an error occurs, this method returns nil and populates the object pointer referenced by error with additional information.

This method applies only to URLs that represent file system resources.

Available in OS X v10.6 and later.

See also:

- 8.9.149 resourceValuesForKeys(keys() as string, targetDelegate as ResourceValuesForKeysDelegateMBS, tag as Variant = nil, PrecacheIcons as boolean = false) 875

8.9.149 resourceValuesForKeys(keys() as string, targetDelegate as ResourceValuesForKeysDelegateMBS, tag as Variant = nil, PrecacheIcons as boolean = false)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Query keys asynchronously.

Notes: The plugin will query values on a different preemptive thread and call your delegate on main thread with results as soon as possible. This way you can keep app responsive while system e.g. loads icons.

For icons, you can set PrecacheIcons to true. In that case plugin will draw icon on the preemptive thread, so icon data is really loaded from disk. When you then draw on main thread, it's really quick.

The delegate has this parameters:

ResourceValuesForKeysDelegateMBS(URL as NSURLMBS, keys() as String, Values as Dictionary, Error as NSErrorMBS, tag as Variant)

See also:

- 8.9.148 resourceValuesForKeys(keys() as string, byref error as NSErrorMBS) as Dictionary 874

8.9.150 `setResourceValue(value as Variant, key as string, byref error as NSErrorMBS) as boolean`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Sets the URL,Äôs resource property for a given key to a given value.

Notes: value: The value for the resource property defined by key.

key: The name of one of the URL,Äôs resource properties.

error: The error that occurred if the resource value could not be set.

Returns true if the resource property named key is successfully set to value; otherwise, false.

This method synchronously writes the new resource value out to disk. Attempts to set a read-only resource property or to set a resource property that is not supported by the resource are ignored and are not considered errors.

If an error occurs, this method returns NO and populates the object pointer referenced by error with additional information.

This method applies only to URLs for file system resources.
Available in OS X v10.6 and later.

Automatic type translation applies (See FAQ about NSDictionary).

8.9.151 `setResourceValues(keyedValues as Dictionary, byref error as NSErrorMBS) as boolean`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Sets the URL,Äôs resource properties for a given set of keys to a given set of values.

Notes: keyedValues: A dictionary of resource values to be set.

error: The error that occurred if one or more resource values could not be set.

Returns true if all resource values in keyedValues are successfully set; otherwise, false.

This method synchronously writes the new resource value out to disk. If an error occurs after some resource properties have been successfully changed, the userInfo dictionary in the returned error object contains a kCFURLKeysOfUnsetValueKey key whose value is an array of the resource values that were not successfully set.

Attempts to set a read-only resource property or to set a resource property that is not supported by the

resource are ignored and are not considered errors.

The order in which the resource values are set is not defined. If you need to guarantee the order in which resource values are set, you should make multiple requests to this method or `setResourceValue(forKey:error:)`.

This method applies only to URLs for file system resources.
Available in OS X v10.6 and later.

Automatic type translation applies (See FAQ about `NSDictionary`).

8.9.152 `setTemporaryResourceValue(value as Variant, key as string)`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Sets a temporary resource value on the URL object.

Notes: value: The value to store.

key: The key where the value should be stored. This key must be unique and must not conflict with any system-defined keys. Reverse-domain-name notation is recommended.

Your app can use a temporary resource value to temporarily store a value for an app-defined resource value key in memory without modifying the actual resource that the URL represents. Once set, you can copy the temporary resource value from the URL object just as you would copy system-defined keys—by calling `getResourceValue` or `resourceValuesForKeys`.

Your app can remove a temporary resource value from the URL object by calling `removeCachedResourceValueForKey` or `removeAllCachedResourceValues` (to remove all temporary values).

This method is applicable only to URLs for file system resources.
Available in OS X v10.9 and later.

8.9.153 `startAccessingSecurityScopedResource` as boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Starts accessing a security scoped bookmark URL.

Notes: Given a `NSURL` created by resolving a bookmark data created with security scope, make the resource referenced by the url accessible to the process. When access to this resource is no longer needed the client must call `stopAccessingSecurityScopedResource`. Each call to `startAccessingSecurityScopedResource` must be balanced with a call to `stopAccessingSecurityScopedResource` (Note: this is not reference counted).

8.9.154 stopAccessingSecurityScopedResource

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Revokes the access granted to the url by a prior successful call to startAccessingSecurityScopedResource.

8.9.155 TagNames as string()

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The names of tags attached to the resource, returned as an array of String values.

Notes: Available in OS X v10.9 and later.

8.9.156 URLByAppendingPathComponent(pathComponent as string) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns a new URL made by appending a path component to the original URL.

Notes: pathComponent: The path component to add to the URL, in its original form (not URL encoded).

Returns a new URL with pathComponent appended.

If the original URL does not end with a forward slash and pathComponent does not begin with a forward slash, a forward slash is inserted between the two parts of the returned URL, unless the original URL is the empty string.

Available in OS X v10.6 and later.

See also:

- 8.9.157 URLByAppendingPathComponent(pathComponent as string, isDirectory as boolean) as NSURLMBS
878

8.9.157 URLByAppendingPathComponent(pathComponent as string, isDirectory as boolean) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns a new URL made by appending a path component to the original URL, along with a trailing slash if the component is designated a directory.

Notes: pathComponent: The path component to add to the URL.

isDirectory: If true, a trailing slash is appended after pathComponent.

Returns a new URL with pathComponent appended.

If the original URL does not end with a forward slash and pathComponent does not begin with a forward slash, a forward slash is inserted between the two parts of the returned URL, unless the original URL is the empty string.

Available in OS X v10.7 and later.

On Mac OS X 10.6 the plugin falls back

See also:

- 8.9.156 URLByAppendingPathComponent(pathComponent as string) as NSURLMBS 878

8.9.158 URLByAppendingPathExtension(PathExtension as string) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns a new URL made by appending a path extension to the original URL.

Notes: pathExtension: The path extension to add to the URL.

Returns a new URL with pathExtension appended.

If the original URL ends with one or more forward slashes, these are removed from the returned URL. A period is inserted between the two parts of the new URL.

Available in OS X v10.6 and later.

8.9.159 URLByDeletingLastPathComponent as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A URL created by taking the receiver and removing the last path component.

Notes: If the receiver,Äôs URL represents the root path, this property contains a copy of the original URL. Otherwise, if the original URL has only one path component, this property contains the empty string.

Available in OS X v10.6 and later.

8.9.160 URLByDeletingPathExtension as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A URL created by taking the receiver and removing the path extension, if any.

Notes: If the receiver represents the root path, this property contains a copy of the original URL. If the URL has multiple path extensions, only the last one is removed.

Available in OS X v10.6 and later.

8.9.161 `URLByResolvingSymlinksInPath` as `NSURLMBS`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A URL that points to the same resource as the receiver and includes no symbolic links.

Notes: If the receiver has no symbolic links, this property contains a copy of the original URL.

If some symbolic links cannot be resolved, this property contains those broken symbolic links.

If the name of the receiving path begins with `/private`, this property strips off the `/private` designator, provided the result is the name of an existing file.

This property only works on URLs with the `file:` path scheme. For all other URLs, it contains a copy of the receiver.

Available in OS X v10.6 and later.

8.9.162 `URLByStandardizingPath` as `NSURLMBS`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A URL that points to the same resource as the original URL using an absolute path.

Notes: This property only works on URLs with the `file:` path scheme. For all other URLs, it returns a copy of the original URL.

Like `stringByStandardizingPath`, this property can make the following changes in the provided URL:

- Expand an initial tilde expression using `stringByExpandingTildeInPath`.
- Reduce empty components and references to the current directory (that is, the sequences `"//"` and `"/./"`) to single path separators.
- In absolute paths only, resolve references to the parent directory (that is, the component `"/."`) to the real parent directory if possible using `stringByResolvingSymlinksInPath`, which consults the file system to resolve each potential symbolic link.

- In relative paths, because symbolic links can,Äôt be resolved, references to the parent directory are left in place.
- Remove an initial component of ”/private” from the path if the result still indicates an existing file or directory (checked by consulting the file system).

Note that the path contained by this property may still have symbolic link components in it. Note also that this property only works with file paths (not, for example, string representations of URLs).

Available in OS X v10.6 and later.

8.9.163 URLsResourceValuesForKeys(URLs() as NSURLMBS, keys() as string, targetDelegate as URLsResourceValuesForKeysDelegateMBS, tag as Variant = nil, PrecacheIcons as boolean = false)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Queries several URLs for values.

Notes:

Similar to resourceValuesForKeys, this method will start a preemptive thread and queries values for all URLs on all keys in background. Once done it calls the delegate on main thread.

For icons, you can set PrecacheIcons to true. In that case plugin will draw icon on the preemptive thread, so icon data is really loaded from disk. When you than draw on main thread, it’s really quick.

the delegate is declared like this:

URLsResourceValuesForKeysDelegateMBS(URLs() as NSURLMBS, keys() as String, Values() as Dictionary, Errors() as NSErrorMBS, tag as Variant)

8.9.164 URLWithHandle(Handle as Integer) as NSURLMBS

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Creates a new URL object based on a handle value.

Notes: Will retain the reference.

8.9.165 URLWithItem(Item as FolderItem) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates an URL based on folderItem.

Example:

```

dim f as FolderItem = GetFolderItem("/Volumes/Ablage1", FolderItem.PathTypeNative)
dim n as NSURLMBS = NSURLMBS.URLWithItem(f)
MsgBox n.absoluteString

```

8.9.166 URLWithString(URL as string) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates and returns an NSURL object initialized with a provided URL string.
See also:

- 8.9.167 URLWithString(URL as string, baseURL as NSURLMBS) as NSURLMBS 882

8.9.167 URLWithString(URL as string, baseURL as NSURLMBS) as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Creates and returns an NSURL object initialized with a base URL and a relative string.

Notes: This method allows you to create a URL relative to a base path or URL. For example, if you have the URL for a folder on disk and the name of a file within that folder, you can construct a URL for the file by providing the folder,Ãs URL as the base path (with a trailing slash) and the filename as the string part.

This method expects URLString to contain only characters that are allowed in a properly formed URL. All other characters must be properly percent escaped. Any percent-escaped characters are interpreted using UTF-8 encoding.

See also:

- 8.9.166 URLWithString(URL as string) as NSURLMBS 882

8.9.168 Properties

8.9.169 absoluteString as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The URL string for the receiver as an absolute URL.

Notes: This property,Ãs value is calculated by resolving the receiver,Ãs string against its base according to the algorithm given in RFC 1808.

(Read only property)

8.9.170 absoluteURL as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: An absolute URL that refers to the same resource as the receiver.

Notes: If the URL is already absolute, this property contains a copy of the receiver. Resolution is performed per RFC 1808.

(Read only property)

8.9.171 AddedToDirectoryDate as Date

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The time at which the resource,Ãs was created or renamed into or within its parent directory, returned as an date.

Notes: Inconsistent behavior may be observed when this attribute is requested on hard-linked items. This property is not supported by all volumes. (read-only)

Available in OS X v10.10.

(Read only property)

8.9.172 AddedToDirectoryDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The time at which the resource,Ãs was created or renamed into or within its parent directory, returned as an date.

Notes: Inconsistent behavior may be observed when this attribute is requested on hard-linked items. This property is not supported by all volumes. (read-only)

Available in OS X v10.10.

(Read only property)

8.9.173 AttributeModificationDate as Date

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Modification date.

Notes: The time at which the resource,Ãs attributes were most recently modified, returned as an NSDate object if the volume supports attribute modification dates, or nil if attribute modification dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.174 `AttributeModificationDateTime` as `DateTime`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Modification date.

Notes: The time at which the resource,Ãs attributes were most recently modified, returned as an `NSDate` object if the volume supports attribute modification dates, or `nil` if attribute modification dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.175 `baseURL` as `NSURLMBS`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The base URL.

Notes: This property contains the base URL. If the receiver is an absolute URL, this property contains `nil`.
(Read only property)

8.9.176 `ContentAccessDate` as `Date`

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The time at which the resource was most recently accessed.

Notes: Returned as date object if the volume supports access dates, or `nil` if access dates are unsupported.
Available in OS X v10.6 and later.
(Read only property)

8.9.177 `ContentAccessDateTime` as `DateTime`

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The time at which the resource was most recently accessed.

Notes: Returned as date object if the volume supports access dates, or `nil` if access dates are unsupported.
Available in OS X v10.6 and later.
(Read only property)

8.9.178 ContentModificationDate as Date

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The modification date.

Notes: The time at which the resource was most recently modified, returned as a date object if the volume supports modification dates, or nil if modification dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.179 ContentModificationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The modification date.

Notes: The time at which the resource was most recently modified, returned as a date object if the volume supports modification dates, or nil if modification dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.180 CreationDate as Date

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The creation date.

Notes: The resource's creation date, returned as an NSDate object if the volume supports creation dates, or nil if creation dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.181 CreationDateTime as DateTime

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: The creation date.

Notes: The resource's creation date, returned as an NSDate object if the volume supports creation dates, or nil if creation dates are unsupported.

Available in OS X v10.6 and later.
(Read only property)

8.9.182 DocumentIdentifier as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The document identifier returned as a number.

Notes: The document identifier is a value assigned by the kernel to a file or directory. This value is used to identify the document regardless of where it is moved on a volume. The identifier persists across system restarts. It is not transferred when the file is copied, but it survives "safe save" operations.

Document identifiers are only unique within a single volume. This property is not supported by all volumes.

Available in OS X v10.10 and iOS 8.0.

(Read only property)

8.9.183 EffectiveIcon as Variant

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's normal icon, returned as an NSImage object.

Notes: Available in OS X v10.6 and later.

Value is a NSImageMBS object.

(Read only property)

8.9.184 filePathURL as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A file path URL that points to the same resource as the URL object.

Notes: If the receiver is a file reference URL, this property contains a copy of the URL converted to a file path URL. If the receiver's URL is a file path URL, this property contains the original URL. If the original URL is not a file URL, or if the resource is not reachable or no longer exists, this property contains nil.

Available in OS X v10.6 and later.

(Read only property)

8.9.185 fileReferenceURL as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns a new file reference URL that points to the same resource as the receiver.

Notes: File reference URLs use a URL path syntax that identifies a file system object by reference, not by

path. This form of file URL remains valid when the file system path of the URL,Ãs underlying resource changes.

If the original URL is a file path URL, this property contains a copy of the URL converted into a file reference URL. If the original URL is a file reference URL, this property contains the original. If the original URL is not a file URL, this property contains nil.

File reference URLs cannot be created to file system objects which do not exist or are not reachable. This property contains nil instead.

In some areas of the file system hierarchy, file reference URLs cannot be generated to the leaf node of the URL path.

A file reference URL's path should never be persistently stored, because it is not valid across system restarts or remounts of volumes. If you need to store a persistent reference to a file system object, use a bookmark instead. You can create a bookmark by calling `bookmarkDataWithOptions`.

Available in OS X v10.6 and later.
(Read only property)

8.9.186 FileResourceIdentifier as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource,Ãs unique identifier, returned as an id.

Notes: This identifier can be used to determine equality between file system resources with the `isEqual:` method. Two resources are equal if they have the same file-system path or if their paths link to the same inode on the same file system.

The value of this identifier is not persistent across system restarts.

Available in OS X v10.7 and later.
Returns hex encoded data.
(Read only property)

8.9.187 FileResourceType as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource,Ãs object type, returned as an NSString object.

Notes: See File Resource Types for possible values.
Available in OS X v10.7 and later.
(Read only property)

8.9.188 `fileSystemRepresentation` as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A string containing the URL,Ãs file system path.
Notes: (Read only property)

8.9.189 `fragment` as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The fragment identifier, conforming to RFC 1808. (read-only)
Notes: This property contains the URL,Ãs fragment. If the receiver does not conform to RFC 1808, this property contains nil. For example, in the URL `http://www.example.com/index.html#jumpLocation`, the fragment identifier is `jumpLocation`.
(Read only property)

8.9.190 `GenerationIdentifier` as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: An opaque generation identifier, returned as an id.
Notes: The generation identifier can be compared using `isEqual` to determine if the data in a document has been modified. For URLs which refer to the same file inode, the generation identifier changes when the data in the file's data fork is changed. Changes to extended attributes or other file system metadata do not change the identifier. For URLs which refer to the same directory inode, the generation identifier changes when direct children of that directory are added, removed or renamed. Changes to the data of the direct children of that directory does not change the generation identifier. The identifier persists across system restarts. It is tied to a specific document on a specific volume and is not transferred when the document is copied to another volume. This property is not supported by all volumes.

Available in OS X v10.10 and iOS 8.0.
(Read only property)

8.9.191 Handle as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: Can be used as NSURL* or CFURLRef for declares.
(Read and Write property)

8.9.192 HasHiddenExtension as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource,Ãs extension is normally removed from its localized name.

Notes: Returned as a Boolean (read-write).
(Read only property)

8.9.193 host as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The host, conforming to RFC 1808. (read-only)

Notes: This property contains the host. For example, in the URL `http://www.example.com/index.html`, the host is `www.example.com`.

If the receiver does not conform to RFC 1808, this property contains nil. The litmus test for conformance to RFC 1808 is as recommended in RFC 1808—specifically, whether the first two characters of resourceSpecifier are slashes (/).

(Read only property)

8.9.194 IsAlias as Boolean

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Whether the file is an alias.

Example:

```
Dim file As folderitem = specialfolder.desktop.child("Hello.txt")
```

```
Dim url As New NSURLMBS(file)  
MsgBox "IsAlias: " + If(url.IsAlias, "yes", "no")
```

Notes: Returns true for alias files.
(Read only property)

8.9.195 IsDirectory as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is a directory.

Notes: Available in OS X v10.6 and later.
(Read only property)

8.9.196 IsExcludedFromBackup as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is excluded from all backups of app data.

Notes: You can use this property to exclude cache and other app support files which are not needed in a backup. Some operations commonly made to user documents cause this property to be reset to false; consequently, do not use this property on user documents.

Available in OS X v10.8 and later.
(Read only property)

8.9.197 IsExecutable as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Key for determining whether the current process (as determined by the EUID) can execute the resource (if it is a file) or search the resource (if it is a directory), returned as a Boolean.

Notes: Available in OS X v10.7 and later.
(Read only property)

8.9.198 isFileReferenceURL as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Returns whether the URL is a file reference URL. S

Notes: (Read only property)

8.9.199 isFileURL as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the scheme is file.

Notes: if myURL.isFileURL is true, then myURL.path is suitable for input into NSFileManager or NSPathUtilities.

(Read only property)

8.9.200 IsHidden as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is normally not displayed to users.

Notes: If the resource is hidden because its name begins with a period, setting this value has no effect.

Available in OS X v10.6 and later.

(Read only property)

8.9.201 IsMountTrigger as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the URL is a file system trigger directory, returned as a Boolean.

Notes: Traversing or opening a file system trigger directory causes an attempt to mount a file system on the directory.

Available in OS X v10.7 and later.

(Read only property)

8.9.202 IsPackage as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is a file package

Notes: A true value means that the resource is a file package.

Available in OS X v10.6 and later.

(Read only property)

8.9.203 IsReadable as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the current process (as determined by the EUID) can read the resource.

Notes: Available in OS X v10.7 and later.

(Read only property)

8.9.204 IsRegularFile as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Determining whether the resource is a regular file, as opposed to a directory or a symbolic link.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.205 IsSymbolicLink as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is a symbolic link.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.206 IsSystemImmutable as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource's system immutable bit is set.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.207 IsUbiquitousItem as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if this item is synced to the cloud, false if it is only a local file.

Notes: (Read only property)

8.9.208 IsUserImmutable as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource's user immutable bit is set.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.209 IsVolume as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the resource is the root directory of a volume.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.210 IsWritable as Boolean

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Whether the current process (as determined by the EUID) can write to the resource.

Notes: Available in OS X v10.7 and later.

(Read only property)

8.9.211 Item as FolderItem

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: For the file URLs the corresponding folderitem.

Notes: (Read only property)

8.9.212 LabelColor as Variant

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's label color, returned as an NSColor object, or nil if the resource has no label color.

Notes: Available in OS X v10.6 and later.

Value is a NSColorMBS object.

(Read only property)

8.9.213 LabelNumber as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's label number.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.214 lastPathComponent as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The last path component. (read-only)

Notes: This property contains the last path component. For example, in the URL file:///path/to/file, the last path component is file.

Available in OS X v10.6 and later.

(Read only property)

8.9.215 LinkCount as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The number of hard links to the resource.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.216 LocalizedLabel as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's localized label text, returned as a string, or empty if the resource has no localized label text.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.217 LocalizedName as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's localized or extension-hidden name, returned as a string.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.218 LocalizedTypeDescription as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's localized type description, returned as a string.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.219 Name as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource's name in the file system, returned as a string.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.220 parameterString as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The parameter string conforming to RFC 1808. (read-only)

Notes: This property contains the parameter string. If the receiver does not conform to RFC 1808, this property contains nil. For example, in the URL file:///path/to/file;foo, the parameter string is foo.

This property should not be confused with the query property, which also often contains a string of parameters.

(Read only property)

8.9.221 ParentDirectoryURL as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The parent directory of the resource, returned as an NSURLMBS object, or nil if the resource is the root directory of its volume.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.222 password as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The password conforming to RFC 1808. (read-only)

Notes: This property contains the password. If the receiver does not conform to RFC 1808, it contains nil. For example, in the URL `http://username:password@www.example.com/index.html`, the password is password.

(Read only property)

8.9.223 path as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The path, conforming to RFC 1808. (read-only)

Notes: This property contains the path, unescaped with the `stringByReplacingPercentEscapesUsingEncoding:` method. If the receiver does not conform to RFC 1808, this property contains nil.

If the receiver contains a file or file reference URL (as determined with `isFileURL`), this property's value is suitable for input into methods of `NSFileManager` or `NSPathUtilities`. If the path has a trailing slash, it is stripped.

If the receiver contains a file reference URL, this property's value provides the current path for the referenced resource, which may be nil if the resource no longer exists.

If the `parameterString` property contains a non-nil value, the path may be incomplete. If the receiver contains an unencoded semicolon, the path property ends at the character before the semicolon. The remainder of the URL is provided in the `parameterString` property.

To obtain the complete path, if `parameterString` contains a non-nil value, append a semicolon, followed by the parameter string.

Per RFC 3986, the leading slash after the authority (host name and port) portion is treated as part of the path. For example, in the URL `http://www.example.com/index.html`, the path is `/index.html`.

Available in OS X v10.8 and later.

(Read only property)

8.9.224 pathExtension as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The path extension. (read-only)

Notes: This property contains the path extension. For example, in the URL `file:///path/to/file.txt`, the path extension is `txt`.

Available in OS X v10.6 and later.
(Read only property)

8.9.225 port as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The port, conforming to RFC 1808. (read-only)

Notes: This property contains the port number. For example, in the URL `http://www.example.com:8080/index.php`, the port number is 8080.

If the receiver does not conform to RFC 1808, this property contains nil. The litmus test for conformance to RFC 1808 is as recommended in RFC 1808—specifically, whether the first two characters of `resourceSpecifier` are slashes (`//`).

(Read only property)

8.9.226 PreferredIOBlockSize as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The preferred block size.

Notes: The optimal block size to use when reading or writing this file’s data, returned as an integer, or zero if the preferred size is not available (read-only).

Available in OS X v10.7 and later.

(Read only property)

8.9.227 QuarantineProperties as Dictionary

Plugin Version: 15.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The quarantine properties as defined in `LSQuarantine.h`.

Example:

```
Dim f As FolderItem = SpecialFolder.UserHome.Child("Downloads").Child("Installation.pdf")
Dim n As New NSURLMBS(f)
Dim d As Dictionary = n.QuarantineProperties
```

Break *// inspect in debugger*

Notes: Available on Mac OS X 10.10 and later.
(Read only property)

8.9.228 query as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The query string, conforming to RFC 1808. (read-only)

Notes: This property contains the query string. If the receiver does not conform to RFC 1808, this property contains nil. For example, in the URL `http://www.example.com/index.php?key1=value1&key2=value2`, the query string is `key1=value1&key2=value2`.

(Read only property)

8.9.229 relativePath as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The relative path, conforming to RFC 1808. (read-only)

Notes: This property contains the relative path of the receiver, Ås URL without resolving against its base URL. If the path has a trailing slash it is stripped. If the receiver is an absolute URL, this property contains the same value as path. If the receiver does not conform to RFC 1808, it contains nil.

(Read only property)

8.9.230 relativeString as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A string representation of the relative portion of the URL. (read-only)

Notes: This property contains a string representation of the relative portion of the URL. If the receiver is an absolute URL this method returns the same value as `absoluteString`.

(Read only property)

8.9.231 resourceSpecifier as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource specifier. (read-only)

Notes: This property contains the resource specifier. For example, in the URL `http://www.example.com/index.html?key1=value1#jumplink`, the resource specifier is `//www.example.com/index.html?key1=value1#jumplink` (everything after the colon).

(Read only property)

8.9.232 `scheme` as `String`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The scheme. (read-only)

Notes: This property contains the scheme. For example, in the URL `http://www.example.com/index.html`, the scheme is `http`.

The full URL is the concatenation of the scheme, a colon (`:`), and the value of `resourceSpecifier`.

The term “protocol” is also sometimes used when talking about network-based URL schemes. However, not all URL schemes are networking protocols—`data://` URLs, for example.

(Read only property)

8.9.233 `standardizedURL` as `NSURLMBS`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: A copy of the URL with any instances of `“.”` or `“”` removed from its path. (read-only)

Notes: This property contains a new `NSURL` object, initialized using the receiver’s path with any instances of `“.”` or `“”` removed.

If the URL conforms to RFC 1808 (the most common form of URL), this property contains the specified URL component; otherwise it contains `nil`. The litmus test for conformance to RFC 1808 is as recommended in RFC 1808—specifically, whether the first two characters of `resourceSpecifier` are slashes (`//`).

(Read only property)

8.9.234 `TypeIdentifier` as `String`

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The resource’s uniform type identifier (UTI), returned as a string.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.235 `UbiquitousItemContainerDisplayName` as `String`

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Returns the name of this item’s container as displayed to users.

Notes: (Read only property)

8.9.236 UbiquitousItemDownloadingError as NSErrorMBS

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Returns the error when downloading the item from iCloud failed,.

Notes: (Read only property)

8.9.237 UbiquitousItemDownloadingStatus as String

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Returns the download status of this item.

Notes: (Read only property)

8.9.238 UbiquitousItemDownloadRequested as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Returns whether a download of this item has already been requested with an API like start-DownloadingUbiquitousItem.

Notes: (Read only property)

8.9.239 UbiquitousItemHasUnresolvedConflicts as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if this item has conflicts outstanding.

Notes: (Read only property)

8.9.240 UbiquitousItemIsDownloaded as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Has never behaved as documented in earlier releases, hence deprecated.

Notes: Equivalent to `UbiquitousItemDownloadingStatus = NSURLUbiquitousItemDownloadingStatusCurrent`.

(Read only property)

8.9.241 UbiquitousItemIsDownloading as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if data is being downloaded for this item.

Notes: (Read only property)

8.9.242 UbiquitousItemIsExcludedFromSync as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if the item is excluded from sync, which means it is locally on disk but won't be available on the server.

Notes: An excluded item is no longer ubiquitous.

(Read only property)

8.9.243 UbiquitousItemIsShared as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if the ubiquitous item is shared.

Notes: (Read only property)

8.9.244 UbiquitousItemIsUploaded as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if there is data present in the cloud for this item.

Notes: (Read only property)

8.9.245 UbiquitousItemIsUploading as Boolean

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: True if data is being uploaded for this item.

Notes: (Read only property)

8.9.246 UbiquitousItemUploadingError as NSErrorMBS

Plugin Version: 22.1, Platform: macOS, Targets: All.

Function: Returns the error when uploading the item to iCloud failed.

Notes: (Read only property)

8.9.247 user as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The user name, conforming to RFC 1808. (read-only)

Notes: This property contains the user name. For example, in the URL ftp://username@www.example.com/, the user name is username.

If the receiver's URL does not conform to RFC 1808, this property returns nil. The litmus test for conformance to RFC 1808 is as recommended in RFC 1808—specifically, whether the first two characters of resourceSpecifier are slashes (/).

(Read only property)

8.9.248 VolumeIdentifier as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The unique identifier of the resource's volume, returned as an id.

Notes: The unique identifier of the resource's volume, returned as an id (read-only).

This identifier can be used with the isEqual method to determine whether two file system resources are on the same volume.

The value of this identifier is not persistent across system restarts.

Available in OS X v10.7 and later.

Returns hex encoded data.

(Read only property)

8.9.249 VolumeURL as NSURLMBS

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The root directory of the resource's volume, returned as an NSURL object.

Notes: Available in OS X v10.6 and later.

(Read only property)

8.9.250 Delegates**8.9.251 ResourceValuesForKeysDelegateMBS(URL as NSURLMBS, keys() as String, Values as Dictionary, Error as NSErrorMBS, tag as variant)**

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The delegate to use for resourceValuesForKeys method.

8.9.252 URLsResourceValuesForKeysDelegateMBS(URLs() as NSURLMBS, keys() as String, Values() as Dictionary, Errors() as NSErrorMBS, tag as variant)

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The delegate to use with URLsResourceValuesForKeys method.

8.10 class NSURLProtectionSpaceMBS

8.10.1 class NSURLProtectionSpaceMBS

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: A class for the Cocoa URL protection space class.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 20.2pr3](#)

8.10.2 Methods

8.10.3 authenticationMethod as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the authentication method to be used for this protection space

8.10.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

8.10.5 host as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the proxy host if this is a proxy authentication, or the host from the URL.

Notes: Returns the host for this protection space.

8.10.6 isProxy as boolean

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Determine if this authenticating protection space is a proxy server

Notes: Returns true if a proxy, false otherwise.

8.10.7 NSURLAuthenticationMethodClientCertificate as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use client certificate authentication for this protection space.

Notes: This authentication method can apply to any protocol.

8.10.8 NSURLAuthenticationMethodDefault as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use the default authentication method for a protocol.

8.10.9 NSURLAuthenticationMethodHTMLForm as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use HTML form authentication for this protection space.

Notes: The URL loading system never issues authentication challenges based on this authentication method. However, if your app authenticates by submitting a web form (or in some other protocol-neutral way), you can specify this protection space when you persist or look up credentials using the `NSURLCredentialStorageMBS` class.

8.10.10 NSURLAuthenticationMethodHTTPBasic as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use HTTP basic authentication for this protection space.

8.10.11 NSURLAuthenticationMethodHTTPEDigest as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use HTTP digest authentication for this protection space.

8.10.12 NSURLAuthenticationMethodNegotiate as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Negotiate whether to use Kerberos or NTLM authentication for this protection space.

8.10.13 NSURLAuthenticationMethodNTLM as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Use NTLM authentication for this protection space.

8.10.14 NSURLAuthenticationMethodServerTrust as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Perform server trust authentication (certificate validation) for this protection space.

Notes: This authentication method can apply to any protocol, and is most commonly used for overriding SSL and TLS chain validation.

8.10.15 NSURLProtectionSpaceFTP as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The protocol type for FTP.

8.10.16 NSURLProtectionSpaceFTPProxy as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The proxy type for FTP proxies.

8.10.17 NSURLProtectionSpaceHTTP as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The protocol type for HTTP.

8.10.18 NSURLProtectionSpaceHTTPProxy as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The proxy type for HTTP proxies.

8.10.19 NSURLProtectionSpaceHTTPS as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The protocol type for HTTPS.

8.10.20 NSURLProtectionSpaceHTTPSPProxy as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The proxy type for HTTPS proxies.

8.10.21 NSURLProtectionSpaceSOCKSPProxy as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: The proxy type for SOCKS proxies.

8.10.22 port as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the proxy port if this is a proxy authentication, or the port from the URL.

Notes: Returns the port for this protection space, or 0 if not set.

8.10.23 protocol as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the protocol of this protection space, if not a proxy.

Notes: Returns the type string, or "" if a proxy.

8.10.24 proxyType as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the type of this protection space, if a proxy.

Notes: Returns the type string, or "" if not a proxy.

8.10.25 realm as string

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Get the authentication realm for which the protection space that needs authentication

Notes: This is generally only available for http authentication, and may be "" otherwise.

8.10.26 receivesCredentialSecurely as boolean

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: Determine if the password for this protection space can be sent securely

Notes: True if a secure authentication method or protocol will be used, false otherwise.

8.10.27 Properties

8.10.28 Handle as Integer

Plugin Version: 7.5, Platform: macOS, Targets: All.

Function: The internal used handle for this class.

Notes: (Read and Write property)

8.11 class NSURLRequestMBS

8.11.1 class NSURLRequestMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: An NSURLRequest object represents a URL load request in a manner independent of protocol and URL scheme.

Example:

```
msgbox HTMLViewer1.mainFrameMBS.DataSource.Request.url
```

Notes: NSURLRequest encapsulates two basic data elements about a URL load request:

The URL to load.

The policy to use when consulting the URL content cache made available by the implementation.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.5](#)
- [MBS Xojo Plugins, version 21.5pr4](#)
- [MBS Releases the MBS Xojo / Real Studio plug-ins in version 16.4](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr4](#)
- [Notes from the last days](#)
- [MBS Real Studio Plugins, version 11.3pr6](#)

Xojo Developer Magazine

- [20.1, page 9: News](#)

8.11.2 Methods

8.11.3 allHTTPHeaderFields as Dictionary

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Returns a dictionary of the HTTP header fields associated with the receiver.

8.11.4 Constructor(url as string)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

Example:

```
dim r as NSURLRequestMBS
r=new NSURLRequestMBS("http://www.apple.com")
```

Notes: Default values are used for cache policy (NSURLRequestUseProtocolCachePolicy) and timeout interval (60 seconds).

On success, handle property is not zero.

See also:

- 8.11.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double) 911

8.11.5 Constructor(url as string, cachePolicy as Integer, timeoutInterval as Double)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

Example:

```
dim r as NSURLRequestMBS
r=new NSURLRequestMBS("http://www.apple.com",NSURLRequestMBS.NSURLRequestReturnCacheDataElseLoad,5.0)
```

Notes: On success, handle property is not zero.

See also:

- 8.11.4 Constructor(url as string) 911

8.11.6 copy as NSURLRequestMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a copy of the request.

Example:

```
// create PUT request
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setHTTPMethod "PUT"
```

```
// make a copy
dim r as NSURLRequestMBS = m.copy

// change first request to POST
m.setHTTPMethod "POST"

// and check values
MsgBox m.HTTPMethod+" "+r.HTTPMethod
```

8.11.7 mutableCopy as NSMutableURLRequestMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates an editable copy of the request.

Example:

```
// create PUT request
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setHTTPMethod "PUT"

// make a copy
dim r as NSMutableURLRequestMBS = m.mutableCopy

// change request to POST
r.setHTTPMethod "POST"

// and check values
MsgBox m.HTTPMethod+" "+r.HTTPMethod
```

8.11.8 requestWithHandle(Handle as Integer) as NSURLRequestMBS

Plugin Version: 16.4, Platform: macOS, Targets: All.

Function: Creates a new request object based on a handle value.

Notes: Will retain the reference.

8.11.9 requestWithURL(url as string) as NSURLRequestMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

Example:

```
dim r as NSURLRequestMBS
```

```
r=r.requestWithURL("http://www.apple.com",0,5.0)
```

Notes: Default values are used for cache policy (NSURLRequestUseProtocolCachePolicy) and timeout interval (60 seconds).

See also:

- 8.11.10 requestWithURL(url as string, cachePolicy as Integer, timeoutInterval as Double) as NSURLRequestMBS 913

8.11.10 requestWithURL(url as string, cachePolicy as Integer, timeoutInterval as Double) as NSURLRequestMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates an NSURLRequest with the given URL.

Example:

```
dim r as NSURLRequestMBS
```

```
r=r.requestWithURL("http://www.apple.com", NSURLRequestMBS.NSURLRequestUseProtocolCachePolicy, 5.0)
```

See also:

- 8.11.9 requestWithURL(url as string) as NSURLRequestMBS 912

8.11.11 valueForHTTPHeaderField(field as string) as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the value of the specified HTTP header field.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setValue("just a test", "test")
MsgBox m.valueForHTTPHeaderField("test")
```

Notes: field: The name of the header field whose value is to be returned. In keeping with the HTTP RFC, HTTP header field names are case-insensitive.

Returns the value associated with the header field field, or "" if there is no corresponding header field.

8.11.12 Properties

8.11.13 attribution as Integer

Plugin Version: 21.5, Platform: macOS, Targets: All.

Function: Returns the NSURLRequestAttribution associated with this request.

Notes: This will return NSURLRequestAttributionDeveloper for requests that have not explicitly set an attribution.

(Read only property)

8.11.14 cachePolicy as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the cache policy of the request.

Example:

```
dim r as NSURLRequestMBS
r=new NSURLRequestMBS("http://www.apple.com",0,5.0)
MsgBox str(r.cachePolicy)
```

Notes: Value is one of the constants in this class.

(Read only property)

8.11.15 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The reference to the internal used NSURLRequest object.

Notes: (Read and Write property)

8.11.16 HTTPBody as memoryblock

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Returns the body of the request.

Notes: This is the data sent in a POST request.

(Read only property)

8.11.17 HTTPMethod as string

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Returns the HTTP method associated with the receiver.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test")
m.setHTTPMethod "PUT"
MsgBox m.HTTPMethod
```

Notes: (Read only property)

8.11.18 HTTPShouldHandleCookies as boolean

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Returns a flag indicating whether this request should use standard cookie handling (sending of cookies with the request and storing any cookies returned in the response).

Notes: (Read only property)

8.11.19 HTTPShouldUsePipelining as boolean

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Reports whether the receiver is not expected to wait for the previous response before transmitting.

Notes: Returns true if the receiver should transmit before the previous response is received. False if the receiver should wait for the previous response before transmitting.

Available in Mac OS X 10.7 or newer.

(Read only property)

8.11.20 isHTTPRequest as boolean

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Whether this is an HTTP request.

Notes: If true, the allHTTPHeaderFields, HTTPShouldHandleCookies, HTTPMethod and HTTPBody methods do work.

(Read only property)

8.11.21 mainDocumentURL as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The main document URL associated with this load.

Example:

```
dim m as new NSMutableURLRequestMBS("http://test.test/test.jpg")
m.setMainDocumentURL "http://test.test/"
MsgBox m.mainDocumentURL
```

Notes: This URL is used for the cookie "same domain as main document" policy. There may also be other future uses.

NOTE: In the current implementation, this value is unused by the webkit framework. A fully functional version of this method will be available in the future.

(Read only property)

8.11.22 networkServiceType as Integer

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns the service type associated with this request.

Notes: This will return NSURLNetworkServiceTypeDefault for requests that have not explicitly set a networkServiceType (using the setNetworkServiceType method). See NSURLNetworkServiceType* constants.

Available in Mac OS X 10.7 or newer.

(Read only property)

8.11.23 timeoutInterval as Double

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the timeout interval of the request.

Example:

```
dim r as NSURLRequestMBS  
  
r=new NSURLRequestMBS("http://www.apple.com",0,5.0)  
  
MsgBox str(r.timeoutInterval)
```

Notes: The timeout interval specifies the limit on the idle interval allotted to a request in the process of loading. The "idle interval" is defined as the period of time that has passed since the last instance of load activity occurred for a request that is in the process of loading. Hence, when an instance of load activity occurs (e.g. bytes are received from the network for a request), the idle interval for a request is reset to 0. If the idle interval ever becomes greater than or equal to the timeout interval, the request is considered to have timed out. This timeout interval is measured in seconds.

(Read only property)

8.11.24 URL as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the URL of the request.

Example:

```
dim r as NSURLRequestMBS  
  
r=new NSURLRequestMBS("http://www.apple.com")  
  
MsgBox r.URL
```

Notes: (Read only property)

8.11.25 Constants

Constants

Constant	Value	Description
NSURLRequestReloadIgnoringCacheData	1	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifically, these constants cover interactions that occur when already-existing cache data is returned to satisfy a URL load request. NSURLRequestReloadIgnoringCacheData Specifies that the existing cache data should be used to satisfy a URL load request. No check of its freshness or validity, should be used to satisfy a URL load request.
NSURLRequestReloadIgnoringLocalAndRemoteCacheData	4	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifies that not only should the local cache data be used, but also other intermediates should be instructed to do so, as the protocol allows. Available in Mac OS X v10.5 and later.
NSURLRequestReloadRevalidatingCacheData	5	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifies that the existing cache data may be used to satisfy a URL load request, but only if it confirms its validity, otherwise the URL is loaded from the origin source. Available in Mac OS X v10.5 and later.
NSURLRequestReturnCacheDataDontLoad	3	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifically, these constants cover interactions that occur when already-existing cache data is returned to satisfy a URL load request. NSURLRequestReturnCacheDataDontLoad Specifies that the existing cache data should be used to satisfy a URL load request, but only if it has an expiration date. However, if there is no existing data to a URL load request, no attempt is made to load from the origin source, and the load is considered to have failed. This is similar to an "offline" mode.
NSURLRequestReturnCacheDataElseLoad	2	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifically, these constants cover interactions that occur when already-existing cache data is returned to satisfy a URL load request. NSURLRequestReturnCacheDataElseLoad Specifies that the existing cache data should be used to satisfy a URL load request, but only if it has an expiration date. However, if there is no existing data to a URL load request, the URL is loaded from the origin source.
NSURLRequestUseProtocolCachePolicy	0	A constant that can be used to specify the type of interaction with the caching system when the URL loading system is used. Specifically, these constants cover interactions that occur when already-existing cache data is returned to satisfy a URL load request. NSURLRequestUseProtocolCachePolicy Specifies that the cache policy defined in the protocol implementation, if any, is used to determine the behavior of the request. This is the default policy for URL load requests.

Network Service Type Constants

Constant	Value	Description
NSURLNetworkServiceTypeBackground	3	Specifies that the request is for background traffic (such as a file download). The service type is used to provide the networking layers a hint of the priority of the request.
NSURLNetworkServiceTypeDefault	0	Is the default value for an NSURLRequest when created. This value should be left unchanged for the vast majority of requests. The service type is used to provide the networking layers a hint of the priority of the request.
NSURLNetworkServiceTypeVideo	2	Specifies that the request is for video traffic. The service type is used to provide the networking layers a hint of the priority of the request.
NSURLNetworkServiceTypeVoice	4	Specifies that the request is for voice data. The service type is used to provide the networking layers a hint of the priority of the request.
NSURLNetworkServiceTypeVoIP	1	Specifies that the request is for voice over IP control traffic. The service type is used to provide the networking layers a hint of the priority of the request.

Attribution		
Constant	Value	Description
NSURLRequestAttributionDeveloper	0	Indicates that the URL was specified by the developer. This is the default value for an NSURLRequest when created.
NSURLRequestAttributionUser	1	Indicates that the URL was specified by the user.

8.12 class NSURLResponseMBS

8.12.1 class NSURLResponseMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: An NSURLResponse object represents a URL load response in a manner independent of protocol and URL scheme.

Notes: NSURLResponse encapsulates the metadata associated with a URL load. Note that NSURLResponse objects do not contain the actual bytes representing the content of a URL.

All methods in this class will catch exceptions from Cocoa and raise a NSErrorMBS instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

8.12.2 Methods

8.12.3 allHeaderFields as Dictionary

Plugin Version: 11.0, Platform: macOS, Targets: All.

Function: Returns all the HTTP header fields of the receiver.

Notes: A dictionary containing all the HTTP header fields of the receiver. By examining this dictionary clients can see the "raw" header information returned by the HTTP server.

8.12.4 Constructor(URL as string, MimeType as string, expectedContentLength as Integer, textEncodingName as string)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initialize an NSURLResponse with the provided values.

Notes: URL: the URL

MIMETYPE: the MIME content type of the response

expectedContentLength: the expected content length of the associated data

textEncodingName: the name of the text encoding for the associated data, if applicable, else "".

Use -1 for an unknown length.

See also FileExtensionToMimeTypeMBS function.

8.12.5 copy as NSURLResponseMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a copy of the object.

Notes: Makes a copy of the RB object and the NSURLResponse object behind.

8.12.6 expectedContentLength as int64

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the expected content length of the response.

Notes: Some protocol implementations report a content length as part of delivering load metadata, but not all protocols guarantee the amount of data that will be delivered in actuality.

Hence, this method returns an expected amount. Clients should use this value as an advisory, and should be prepared to deal with either more or less data.

Returns the expected content length of the receiver, or -1 if there is no expectation that can be arrived at regarding expected content length.

8.12.7 isHTTPResponse as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Whether this response is a http response.

Notes: the StatusCode property can only be used on HTTP Responses.

8.12.8 localizedStringForStatusCode(statusCode as Integer) as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Convenience method which returns a localized string corresponding to the status code for this response.

Notes: statusCode: the status code to use to produce a localized string.

Returns a localized string corresponding to the given status code or an empty string.

8.12.9 MIMEType as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the MIME type of the response.

Example:

`msgbox HTMLViewer1.mainFrameMBS.DataSource.response.MIMETYPE`

Notes: The MIME type is based on the information provided from an origin source. However, that value may be changed or corrected by a protocol implementation if it can be determined that the origin server or source reported the information incorrectly or imprecisely. An attempt to guess the MIME type may be made if the origin source did not report any such information.

See also `FileExtensionToMimeTypeMBS` function.

8.12.10 `statusCode` as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the HTTP status code of the response.

Notes: Returns -1 if no statuscode is available (e.g. in case this is no ta http response)

See also: `localizedStringForStatusCode`

8.12.11 `suggestedFilename` as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns a suggested filename if the resource were saved to disk.

Notes: The method first checks if the server has specified a filename using the content disposition header. If no valid filename is specified using that mechanism, this method checks the last path component of the URL. If no valid filename can be obtained using the last path component, this method uses the URL's host as the filename.

If the URL's host can't be converted to a valid filename, the filename "unknown" is used. In those cases, this method appends the proper file extension based on the MIME type. This method always returns a valid filename.

8.12.12 `textEncodingName` as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the name of the text encoding of the response.

Notes: This name will be the actual string reported by the origin source during the course of performing a protocol-specific URL load. Clients can inspect this string and convert it to a `TextEncoding` using the methods and functions made available in the appropriate framework.

Returns the name of the text encoding of the response, or an empty string if no text encoding was specified.

8.12.13 URL as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Returns the URL of the response.

Example:

```
msgbox HTMLViewer1.mainFrameMBS.DataSource.response.URL
```

8.12.14 Properties

8.12.15 Handle as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The internal used reference to the NSURLResponse object.

Notes: (Read and Write property)

8.12.16 Constants

Constants

Constant	Value	Description
NSURLResponseUnknownLength	-1	The constant for use with expectedContentLength for an unknown length.

Chapter 9

Cocoa Text

9.1 class NSAttributedStringMBS

9.1.1 class NSAttributedStringMBS

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: A class to represent a string in the Cocoa world with attached attributes.

Example:

```
dim a as new NSAttributedStringMBS
```

```
if a.initWithHTML("<B>Hello</B>") then  
MsgBox a.htmlString  
end if
```

Blog Entries

- [RTF functions in MBS Plugins](#)
- [News from the MBS Xojo Plugins Version 22.2](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.0](#)
- [News from the MBS Xojo Plugins in version 21.5](#)
- [Styled Text for Labels in your Xojo iOS app](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.2](#)
- [MBS Xojo plug-ins in version 16.0](#)
- [Tip of the day: Adding links to Textarea on OS X](#)

- [Using NSTextViewMBS](#)
- [MBS Releases the MBS Real Studio plug-ins in version 12.0](#)

Xojo Developer Magazine

- [5.6, page 32: Third Party Plugins: Statusitems, Getting an icon in the top right of the Mac OS X menubar by Christian Schmitz](#)
- [20.2, page 10: News](#)
- [19.4, page 10: News](#)
- [10.3, page 9: News](#)

9.1.2 Methods

9.1.3 AsCFAttributedString as Variant

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new CFAttributedStringMBS object pointing to same attributed string.

Example:

```
// make NS version
dim n as new NSAttributedStringMBS

if n.initWithString("Hello World") then
  // convert
  dim c as CFAttributedStringMBS = n.AsCFAttributedString

  // and check content
  MsgBox c.String
end if
```

Notes: For passing to functions which need a CFAttributedStringMBS.

9.1.4 attributeAtIndex(name as string, location as UInt64) as Variant

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the value for an attribute with a given name of the character at a given index, and by reference the range over which the attribute applies.

Notes: name: The name of an attribute.

location: The index for which to return attributes. This value must not exceed the bounds of the receiver.

`effectiveRange`: Optional. If the named attribute exists at `index`, upon return `aRange` contains a range over which the named attribute's value applies. If the named attribute does not exist at `index`, upon return `aRange` contains the range over which the attribute does not exist.

The range isn't necessarily the maximum range covered by `attributeName`, and its extent is implementation-dependent. If you need the maximum range, use the other variant of this method.

Returns the value for the attribute named `attributeName` of the character at `index`, or `nil` if there is no such attribute.

Raises an `NSRangeException` if `index` lies beyond the end of the receiver's characters.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.5 `attributeAtIndex(name as string, location as UInt64, inRange as NSRangeMBS)` as Variant 927

9.1.5 `attributeAtIndex(name as string, location as UInt64, inRange as NSRangeMBS)` as Variant

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the value for the attribute with a given name of the character at a given index, and by reference the range over which the attribute applies.

Notes: `name`: The name of an attribute.

`location`: The index at which to test for `attributeName`.

`longestEffectiveRange`: Optional. If the named attribute exists at `index`, upon return `aRange` contains the full range over which the value of the named attribute is the same as that at `index`, clipped to `rangeLimit`. If the named attribute does not exist at `index`, upon return `aRange` contains the full range over which the attribute does not exist, clipped to `rangeLimit`.

`inRange`: The range over which to search for continuous presence of `attributeName`. This value must not exceed the bounds of the receiver.

Returns the value for the attribute named `attributeName` of the character at `index`, or `nil` if there is no such attribute.

Raises an `NSRangeException` if `index` or any part of `rangeLimit` lies beyond the end of the receiver's characters.

If you don't need the longest effective range, it's far more efficient to use the other variant method to retrieve the attribute value.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.4 `attributeAtIndex(name as string, location as UInt64) as Variant`

926

9.1.6 `attributeAtIndex2(name as string, location as UInt64, byref effectiveRange as NSRangeMBS) as Variant`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the value for an attribute with a given name of the character at a given index, and by reference the range over which the attribute applies.

Example:

```
// list all links
Dim n As NSAttributedStringMBS = TextArea1.NSTextViewMBS.textStorage

Dim pos As Integer = 0
Dim Len As Integer = n.Length
Dim effectiveRange As NSRangeMBS

While pos < Len

    Dim v As Variant = n.attributeAtIndex2(n.NSLinkAttributeName, pos, effectiveRange)

    If v <> Nil Then
        Dim url As String = v
        ListBox1.AddRow url
    End If

    pos = pos + effectiveRange.Length

Wend
```

Notes: name: The name of an attribute.

location: The index for which to return attributes. This value must not exceed the bounds of the receiver.

effectiveRange: Optional. If the named attribute exists at index, upon return aRange contains a range over which the named attribute's value applies. If the named attribute does not exist at index, upon return aRange contains the range over which the attribute does not exist.

The range isn't necessarily the maximum range covered by attributeName, and its extent is implementation-dependent. If you need the maximum range, use the other variant of this method.

Returns the value for the attribute named attributeName of the character at index index, or nil if there is no such attribute.

Raises an NSRangeException if index lies beyond the end of the receiver's characters.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.7 `attributeAtIndex2(name as string, location as UInt64, byref longestEffectiveRange as NSRangeMBS, inRange as NSRangeMBS)` as Variant 929

9.1.7 `attributeAtIndex2(name as string, location as UInt64, byref longestEffectiveRange as NSRangeMBS, inRange as NSRangeMBS)` as Variant

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the value for the attribute with a given name of the character at a given index, and by reference the range over which the attribute applies.

Notes: name: The name of an attribute.

location: The index at which to test for attributeName.

longestEffectiveRange: Optional. If the named attribute exists at index, upon return aRange contains the full range over which the value of the named attribute is the same as that at index, clipped to rangeLimit. If the named attribute does not exist at index, upon return aRange contains the full range over which the attribute does not exist, clipped to rangeLimit.

inRange: The range over which to search for continuous presence of attributeName. This value must not exceed the bounds of the receiver.

Returns the value for the attribute named attributeName of the character at index, or nil if there is no such attribute.

Raises an NSRangeException if index or any part of rangeLimit lies beyond the end of the receiver's characters.

If you don't need the longest effective range, it's far more efficient to use the other variant method to retrieve the attribute value.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.6 `attributeAtIndex2(name as string, location as UInt64, byref effectiveRange as NSRangeMBS)` as Variant 928

9.1.8 `attributedStringWithAttachment(attachment as NSTextAttachmentMBS) as NSAttributedStringMBS`

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Creates an attributed string with an attachment.

Example:

```
dim content as MemoryBlock = "Hello World"
```

```

dim f as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(content)
f.filename = "HelloWorld.txt"

dim a as new NSTextAttachmentMBS(f)
dim s as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(a)

```

Notes: This is a convenience method for creating an attributed string containing an attachment using `NSAttachmentCharacter` as the base character.

9.1.9 attributedStringWithAttributedString(text as NSAttributedStringMBS) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of the given attributed string.

9.1.10 attributedStringWithDocFormat(data as memoryblock) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes string with content of given DOC file data.

Notes: `documentAttributes`: Optional dictionary to receive the attributes.

See also:

- 9.1.11 `attributedStringWithDocFormat(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS` 930

9.1.11 attributedStringWithDocFormat(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes string with content of given DOC file data.

Notes: `documentAttributes`: Optional dictionary to receive the attributes.

See also:

- 9.1.10 `attributedStringWithDocFormat(data as memoryblock) as NSAttributedStringMBS` 930

9.1.12 attributedStringWithHTML(data as memoryblock) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.13 attributedStringWithHTML(data as memoryblock, BaseURL as string) as NSAttributedStringMBS 931
- 9.1.14 attributedStringWithHTML(data as memoryblock, BaseURL as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 931
- 9.1.15 attributedStringWithHTML(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 932

9.1.13 attributedStringWithHTML(data as memoryblock, BaseURL as string) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes the object with html code from a given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.12 attributedStringWithHTML(data as memoryblock) as NSAttributedStringMBS 931
- 9.1.14 attributedStringWithHTML(data as memoryblock, BaseURL as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 931
- 9.1.15 attributedStringWithHTML(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 932

9.1.14 attributedStringWithHTML(data as memoryblock, BaseURL as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes the object with html code from a given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.12 attributedStringWithHTML(data as memoryblock) as NSAttributedStringMBS 931
- 9.1.13 attributedStringWithHTML(data as memoryblock, BaseURL as string) as NSAttributedStringMBS 931
- 9.1.15 attributedStringWithHTML(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 932

9.1.15 attributedStringWithHTML(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.12 attributedStringWithHTML(data as memoryblock) as NSAttributedStringMBS 931
- 9.1.13 attributedStringWithHTML(data as memoryblock, BaseURL as string) as NSAttributedStringMBS 931
- 9.1.14 attributedStringWithHTML(data as memoryblock, BaseURL as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 931

9.1.16 attributedStringWithHTMLOld(data as string) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Notes: With plugin version 9.4 the attributedStringWithHTML method uses the system function to parse html. The old plugin function is available with the name attributedStringWithHTMLOld.

9.1.17 attributedStringWithPath(file as folderitem) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Example:

```
// load rtf file into textarea
dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim n as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithPath(file)
dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
t.textStorage.setAttributedString(n)
```

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.18 attributedStringWithPath(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933
- 9.1.19 attributedStringWithPath(path as string) as NSAttributedStringMBS 933

9.1. CLASS NSATTRIBUTEDSTRINGMBS 933

- 9.1.20 attributedStringWithPath(path as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933

9.1.18 attributedStringWithPath(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.17 attributedStringWithPath(file as folderitem) as NSAttributedStringMBS 932
- 9.1.19 attributedStringWithPath(path as string) as NSAttributedStringMBS 933
- 9.1.20 attributedStringWithPath(path as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933

9.1.19 attributedStringWithPath(path as string) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given path string.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.17 attributedStringWithPath(file as folderitem) as NSAttributedStringMBS 932
- 9.1.18 attributedStringWithPath(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933
- 9.1.20 attributedStringWithPath(path as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933

9.1.20 attributedStringWithPath(path as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given path string.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.17 attributedStringWithPath(file as folderitem) as NSAttributedStringMBS 932

- 9.1.18 attributedStringWithPath(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 933
- 9.1.19 attributedStringWithPath(path as string) as NSAttributedStringMBS 933

9.1.21 attributedStringWithRTF(data as memoryblock) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTF file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.22 attributedStringWithRTF(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 934

9.1.22 attributedStringWithRTF(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTF file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.21 attributedStringWithRTF(data as memoryblock) as NSAttributedStringMBS 934

9.1.23 attributedStringWithRTFD(data as memoryblock) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTFD file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.24 attributedStringWithRTFD(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 934

9.1.24 attributedStringWithRTFD(data as memoryblock, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTFD file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

9.1. CLASS NSATTRIBUTEDSTRINGMBS 935

- 9.1.23 attributedStringWithRTFD(data as memoryblock) as NSAttributedStringMBS 934

9.1.25 attributedStringWithString(text as string) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given plain text.

Example:

```
// create Hello World in red
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("Hello World")
dim m as NSMutableAttributedStringMBS = a.mutableCopy

m.addAttribute(a.NSForegroundColorAttributeName, NSColorMBS.redColor, new NSRangeMBS(0, m.length))

// put it in a textarea
TextArea1.NSTextViewMBS.textStorage.setAttributedString m
```

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.26 attributedStringWithString(text as string, withAttributes as dictionary) as NSAttributedStringMBS 935

9.1.26 attributedStringWithString(text as string, withAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given plain text.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.25 attributedStringWithString(text as string) as NSAttributedStringMBS 935

9.1.27 attributedStringWithURL(file as folderitem) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.28 attributedStringWithURL(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936

- 9.1.29 attributedStringWithURL(url as string) as NSAttributedStringMBS 936
- 9.1.30 attributedStringWithURL(url as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936

9.1.28 attributedStringWithURL(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.27 attributedStringWithURL(file as folderitem) as NSAttributedStringMBS 935
- 9.1.29 attributedStringWithURL(url as string) as NSAttributedStringMBS 936
- 9.1.30 attributedStringWithURL(url as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936

9.1.29 attributedStringWithURL(url as string) as NSAttributedStringMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.27 attributedStringWithURL(file as folderitem) as NSAttributedStringMBS 935
- 9.1.28 attributedStringWithURL(file as folderitem, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936
- 9.1.30 attributedStringWithURL(url as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS 936

9.1.30 attributedStringWithURL(url as string, byref DocumentAttributes as dictionary) as NSAttributedStringMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.27 attributedStringWithURL(file as folderitem) as NSAttributedStringMBS 935

9.1. CLASS *NSATTRIBUTEDSTRINGMBS* 937

- 9.1.28 `attributedStringWithURL(file as folderitem, byref DocumentAttributes as dictionary)` as `NSAttributedStringMBS` 936
- 9.1.29 `attributedStringWithURL(url as string)` as `NSAttributedStringMBS` 936

9.1.31 `attributedStringFromRange(range as NSRangeMBS)` as `NSAttributedStringMBS`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns an `NSAttributedString` object consisting of the characters and attributes within a given range in the receiver.

Notes: Range: The range from which to create a new attributed string. `aRange` must lie within the bounds of the receiver.

Returns an `NSAttributedString` object consisting of the characters and attributes within `aRange` in the receiver.

Raises an `NSRangeException` if any part of `aRange` lies beyond the end of the receiver's characters. This method treats the length of the string as a valid range value that returns an empty string.

9.1.32 `attributesAtIndex(location as UInt64)` as dictionary

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the attributes for the character at a given index.

Notes: location: The index for which to return attributes. This value must lie within the bounds of the receiver.

Range: Optional. Upon return, the range over which the attributes and values are the same as those at index. This range isn't necessarily the maximum range covered, and its extent is implementation-dependent. If you need the maximum range, use other variant of this method.

Returns the attributes for the character at index.

Raises an `NSRangeException` if index lies beyond the end of the receiver's characters.

See also:

- 9.1.33 `attributesAtIndex(location as UInt64, inRange as NSRangeMBS)` as dictionary 937

9.1.33 `attributesAtIndex(location as UInt64, inRange as NSRangeMBS)` as dictionary

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the attributes for the character at a given index, and by reference the range over which the attributes apply.

Notes: location: The index for which to return attributes. This value must not exceed the bounds of the receiver.

range: Optional. Upon return contains the maximum range over which the attributes and values are the same as those at index, clipped to range Limit.

inRange: The range over which to search for continuous presence of the attributes at index. This value must not exceed the bounds of the receiver.

Raises an NSRangeException if index or any part of rangeLimit lies beyond the end of the receiver's characters.

If you don't need the range information, it's far more efficient to use the other variant of this method to retrieve the attribute value.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.32 attributesAtIndex(location as UInt64) as dictionary 937

9.1.34 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS) as dictionary

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the attributes for the character at a given index.

Notes: location: The index for which to return attributes. This value must lie within the bounds of the receiver.

Range: Optional. Upon return, the range over which the attributes and values are the same as those at index. This range isn't necessarily the maximum range covered, and its extent is implementation-dependent.

If you need the maximum range, use other variant of this method.

Returns the attributes for the character at index.

Raises an NSRangeException if index lies beyond the end of the receiver's characters.

See also:

- 9.1.35 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS, inRange as NSRangeMBS) as dictionary 938

9.1.35 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS, inRange as NSRangeMBS) as dictionary

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Returns the attributes for the character at a given index, and by reference the range over which the attributes apply.

Notes: location: The index for which to return attributes. This value must not exceed the bounds of the receiver.

range: Optional. Upon return contains the maximum range over which the attributes and values are the same as those at index, clipped to range Limit.

inRange: The range over which to search for continuous presence of the attributes at index. This value must not exceed the bounds of the receiver.

Raises an NSRangeException if index or any part of rangeLimit lies beyond the end of the receiver's characters.

If you don't need the range information, it's far more efficient to use the other variant of this method to retrieve the attribute value.

For information about where to find the attribute keys for the returned dictionary, see the overview section of this document.

See also:

- 9.1.34 attributesAtIndex2(location as UInt64, byref range as NSRangeMBS) as dictionary 938

9.1.36 Constructor

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The constructor to create empty attributed string.

9.1.37 Convert_Operator as string

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

Function: Converts to string by just returning string part.

9.1.38 copy as NSAttributedStringMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a copy of the attributed string.

9.1.39 CopyToClipboard as Boolean

Plugin Version: 18.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Copies styled text to clipboard.

9.1.40 `dataFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil, byref error as NSErrorMBS) as memoryblock`

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

Function: Returns an data object that contains a text stream corresponding to the characters and attributes within the given range.

Notes: offset and length: The range.

documentAttributes: A required dictionary specifying the document attributes. The dictionary contains values from Document Types and must at least contain `NSDocumentTypeDocumentAttribute`.

error: An in-out variable containing an encountered error, if any.

Returns the data for the attributed string, or nil if failure. When nil, error encapsulates the error information.

Raises an `NSRangeException` if any part of range lies beyond the end of the receiver,Ãs characters.

9.1.41 `docFormatFromRange(documentAttributes as dictionary = nil) as MemoryBlock`

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates DOC data from the whole string.

Notes: Same as `docFormatFromRange(0,length)`

Returns nil on failure.

documentAttributes can optionally be a dictionary with document attributes like author or title.

See also:

- 9.1.42 `docFormatFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock` 940

9.1.42 `docFormatFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock`

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates DOC data from the current string range.

Notes: Returns nil on failure.

documentAttributes can optionally be a dictionary with document attributes like author or title.

See also:

- 9.1.41 `docFormatFromRange(documentAttributes as dictionary = nil) as MemoryBlock` 940

9.1.43 fileWrapperFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil, byref Error as NSErrorMBS) as NSFileWrapperMBS

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

Function: Returns an NSFileWrapper object that contains a text stream corresponding to the characters and attributes within the given range.

Example:

```
// get styled text from htmlviewer
dim w as WebViewMBS = HTMLViewer1.WebViewMBS
dim f as WebFrameMBS = w.mainFrame
dim v as WebFrameViewMBS = f.frameView
dim d as WebDocumentViewMBS = v.documentView
dim n as NSAttributedStringMBS = d.attributedString

// package it
dim da as new Dictionary
da.Value(n.NSDocumentTypeDocumentAttribute) = n.NSRTFDTextDocumentType

dim e as NSErrorMBS
dim fw as NSFileWrapperMBS = n.fileWrapperFromRange(0, n.Length, da, e)
if e <> nil then
  MsgBox e.LocalizedDescription
  Return
end if

// write to disk
dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
if fw.writeToFile(file, e) then
  MsgBox "OK"
else
  MsgBox e.LocalizedDescription
end if
```

Notes: offset and length: The range.

documentAttributes: A required dictionary specifying the document attributes. The dictionary contains values from Document Types and must at least contain NSDocumentTypeDocumentAttribute.

error: An in-out variable containing an encountered error, if any.

Returns a file wrapper for the appropriate document type, or nil if failure. When nil, error encapsulates the error information.

Raises an NSRangeException if any part of range lies beyond the end of the receiver,Ã characters.

9.1.44 FromClipboard as NSAttributedStringMBS

Plugin Version: 18.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries attributed string from clipboard.

9.1.45 GeneratePDF(PrintOptions as Variant = nil) as MemoryBlock

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a PDF for attributed string.

Example:

```
// read file
dim fi as FolderItem = SpecialFolder.Desktop.Child("test.docx")

dim d as new Dictionary
dim n as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithPath(fi, d)

// write pdf
dim p as MemoryBlock = n.GeneratePDF(nil)

dim fo as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
dim bo as BinaryStream = BinaryStream.Create(fo)
bo.Write p
```

Notes: PrintOptions is optional NSPrintInfoMBS object for print settings like margin.

9.1.46 htmlString as string

Plugin Version: 7.8, Platform: macOS, Targets: All.

Function: A self made function to return the text content as html string preserving the style information.

Example:

```
dim s as NSAttributedStringMBS

s=new NSAttributedStringMBS

if s.initwithhtml("<b>Hello</b>") then
```

```
MsgBox s.htmlstring // shows "<b>Hello</b>"
MsgBox s.text // shows "Hello"
end if
```

Notes: Does not always work well, but can help.
(Apple has no official function for this)

9.1.47 initWithAttributedString(text as NSAttributedStringMBS) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of the given attributed string.

Notes: Returns true on success.

9.1.48 initWithDocFormat(data as MemoryBlock) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes string with content of given DOC file data.

Notes: Returns true on success.

documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.49 initWithDocFormat(data as memoryblock, byref documentAttributes as dictionary) as boolean
943

9.1.49 initWithDocFormat(data as memoryblock, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes string with content of given DOC file data.

Notes: Returns true on success.

documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.48 initWithDocFormat(data as MemoryBlock) as boolean 943

9.1.50 initWithHTML(data as MemoryBlock) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Example:

```
dim n as NSAttributedStringMBS
n=new NSAttributedStringMBS
if n.initWithHTML("<b>test</b>") then
MsgBox n.text
else
MsgBox "failed"
end if

dim attribText as new NSAttributedStringMBS
dim text as string = "<P>Hello √/§/°</P>"

text = ConvertEncoding(text, encodings.ISOLatin1)

If attribText.initWithHTML(text) Then
MsgBox attribText.text
End If
```

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

With plugin version 9.4 this uses the system function to parse html. The old plugin function is available with the name initWithHTMLOld.

On Mac OS X 10.6 the text encoding expected is ISO Latin 1 as far as I see.

See also:

- 9.1.51 initWithHTML(data as MemoryBlock, BaseURL as string) as boolean 944
- 9.1.52 initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean 945
- 9.1.53 initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean 945
- 9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean 946

9.1.51 initWithHTML(data as MemoryBlock, BaseURL as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes the object with html code from a given url.

Example:

```

dim n as NSAttributedStringMBS
n=new NSAttributedStringMBS
if n.initWithHTML("<b>test</b>","http://www.apple.com") then
MsgBox n.text
else
MsgBox "failed"
end if

```

Notes: Returns true on success.

documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.50 initWithHTML(data as MemoryBlock) as boolean 943
- 9.1.52 initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean 945
- 9.1.53 initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean 945
- 9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean 946

9.1.52 initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes the object with html code from a given url.

Notes: Returns true on success.

documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.50 initWithHTML(data as MemoryBlock) as boolean 943
- 9.1.51 initWithHTML(data as MemoryBlock, BaseURL as string) as boolean 944
- 9.1.53 initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean 945
- 9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean 946

9.1.53 initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Notes: Returns true on success.

documentAttributes: Optional dictionary to receive the attributes.

With plugin version 9.4 this uses the system function to parse html. The old plugin function is available with the name initWithHTMLOld.

On Mac OS X 10.6 the text encoding expected is ISO Latin 1 as far as I see.

See also:

- 9.1.50 initWithHTML(data as MemoryBlock) as boolean 943
- 9.1.51 initWithHTML(data as MemoryBlock, BaseURL as string) as boolean 944
- 9.1.52 initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean 945
- 9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean 946

9.1.54 initWithHTML(data as memoryblock, options as Dictionary, byref documentAttributes as dictionary) as boolean

Plugin Version: 22.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Example:

```
Dim theBody As String = "<html><body>Test √§√√°</body></html>"
```

```
Dim documentAttributes As Dictionary
```

```
Dim documentOptions As New Dictionary
```

```
// define the text encoding
```

```
Const NSUTF8StringEncoding = 4
```

```
documentOptions.Value(NSAttributedStringMBS.NSCharacterEncodingDocumentOption) = NSUTF8StringEncoding
```

```
// and a default base URL for finding images
```

```
documentOptions.Value(NSAttributedStringMBS.NSBaseURLDocumentOption) = NSURLMBS.URLWithString("https://monkeybreadsoftware.de/")
```

```
// and a timeout for network queries
```

```
documentOptions.Value(NSAttributedStringMBS.NSTimeoutDocumentOption) = 30
```

```
Dim BodyAttributed As New NSAttributedStringMBS
```

```
If BodyAttributed.initWithHTML(theBody, documentOptions, documentAttributes) Then
```

```
    MessageBox BodyAttributed.Text
```

```
Else
```

```
Break // failed?
End If
```

Notes: Returns true on success.

options: Put in some options for parsing the document like `NSCharacterEncodingDocumentOption` or `NS-TextEncodingNameDocumentOption`.

documentAttributes: Optional dictionary to receive the attributes.

See also:

- 9.1.50 `initWithHTML(data as MemoryBlock) as boolean` 943
- 9.1.51 `initWithHTML(data as MemoryBlock, BaseURL as string) as boolean` 944
- 9.1.52 `initWithHTML(data as memoryblock, BaseURL as string, byref documentAttributes as dictionary) as boolean` 945
- 9.1.53 `initWithHTML(data as memoryblock, byref documentAttributes as dictionary) as boolean` 945

9.1.55 `initWithHTMLOld(data as string) as boolean`

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Initializes string with content of given HTML file data.

Notes: With plugin version 9.4 the `initWithHTML` method uses the system function to parse html. The old plugin function is available with the name `initWithHTMLOld`.

9.1.56 `initWithPath(file as folderitem) as boolean`

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.57 `initWithPath(file as folderitem, byref documentAttributes as dictionary) as boolean` 947
- 9.1.58 `initWithPath(path as string) as boolean` 948
- 9.1.59 `initWithPath(path as string, byref documentAttributes as dictionary) as boolean` 948

9.1.57 `initWithPath(file as folderitem, byref documentAttributes as dictionary) as boolean`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.56 initWithPath(file as folderitem) as boolean 947
- 9.1.58 initWithPath(path as string) as boolean 948
- 9.1.59 initWithPath(path as string, byref documentAttributes as dictionary) as boolean 948

9.1.58 initWithPath(path as string) as boolean

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given path string.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.56 initWithPath(file as folderitem) as boolean 947
- 9.1.57 initWithPath(file as folderitem, byref documentAttributes as dictionary) as boolean 947
- 9.1.59 initWithPath(path as string, byref documentAttributes as dictionary) as boolean 948

9.1.59 initWithPath(path as string, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given path string.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.56 initWithPath(file as folderitem) as boolean 947
- 9.1.57 initWithPath(file as folderitem, byref documentAttributes as dictionary) as boolean 947
- 9.1.58 initWithPath(path as string) as boolean 948

9.1.60 initWithRTF(data as MemoryBlock) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTF file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.61 initWithRTF(data as memoryblock, byref documentAttributes as dictionary) as boolean 949

9.1.61 initWithRTF(data as memoryblock, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTF file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.60 initWithRTF(data as MemoryBlock) as boolean 948

9.1.62 initWithRTFD(data as MemoryBlock) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTFD file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.63 initWithRTFD(data as memoryblock, byref documentAttributes as dictionary) as boolean 949

9.1.63 initWithRTFD(data as memoryblock, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given RTFD file data.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.62 initWithRTFD(data as MemoryBlock) as boolean 949

9.1.64 initWithString(text as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of given plain text.

Example:

```
dim n as NSAttributedStringMBS
n=new NSAttributedStringMBS
if n.initWithHTML("test") then
MsgBox n.text
else
MsgBox "failed"
end if
```

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.65 initWithString(text as string, withAttributes as Dictionary) as boolean

950

9.1.65 initWithString(text as string, withAttributes as Dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of given plain text.

Example:

```
dim n as NSAttributedStringMBS
n=new NSAttributedStringMBS
if n.initWithHTML("test") then
MsgBox n.text
else
MsgBox "failed"
end if
```

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.64 initWithString(text as string) as boolean

949

9.1.66 initWithURL(file as folderitem) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Example:

```

dim n as NSAttributedStringMBS
n=new NSAttributedStringMBS
if n.initWithURL(SpecialFolder.Desktop.Child("testfile")) then
MsgBox n.text
else
MsgBox "failed"
end if

```

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.67 initWithURL(file as folderitem, byref documentAttributes as dictionary) as boolean 951
- 9.1.68 initWithURL(url as string) as boolean 952
- 9.1.69 initWithURL(url as string, byref documentAttributes as dictionary) as boolean 952

9.1.67 initWithURL(file as folderitem, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given file.

Example:

```

dim n as NSAttributedStringMBS
dim documentAttributes as dictionary
n=new NSAttributedStringMBS
if n.initWithURL(SpecialFolder.Desktop.Child("testfile"), documentAttributes) then
MsgBox n.text
else
MsgBox "failed"
end if

```

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.66 initWithURL(file as folderitem) as boolean 950
- 9.1.68 initWithURL(url as string) as boolean 952
- 9.1.69 initWithURL(url as string, byref documentAttributes as dictionary) as boolean 952

9.1.68 initWithURL(url as string) as boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.66 initWithURL(file as folderitem) as boolean 950
- 9.1.67 initWithURL(file as folderitem, byref documentAttributes as dictionary) as boolean 951
- 9.1.69 initWithURL(url as string, byref documentAttributes as dictionary) as boolean 952

9.1.69 initWithURL(url as string, byref documentAttributes as dictionary) as boolean

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Initializes string with content of file at the given url.

Notes: documentAttributes: Optional dictionary to receive the attributes.

Returns true on success.

See also:

- 9.1.66 initWithURL(file as folderitem) as boolean 950
- 9.1.67 initWithURL(file as folderitem, byref documentAttributes as dictionary) as boolean 951
- 9.1.68 initWithURL(url as string) as boolean 952

9.1.70 isEqualToAttributedString(other as NSAttributedStringMBS) as Boolean

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Compares both strings.

Example:

```
dim n as NSAttributedStringMBS
dim m as NSAttributedStringMBS
```

```
n=new NSAttributedStringMBS
if n.initWithString("test") then
  MsgBox n.text
else
  MsgBox "failed"
end if
```

```
m=new NSAttributedStringMBS
if m.initWithString("test") then
  MsgBox n.text
else
  MsgBox "failed"
end if

if m.isEqualToAttributedString(n) then
  MsgBox "Ok"
else
  MsgBox "failed"
end if

m=new NSAttributedStringMBS
if m.initWithString("test2") then
  MsgBox n.text
else
  MsgBox "failed"
end if

if m.isEqualToAttributedString(n) then
  MsgBox "failed"
else
  MsgBox "Ok"
end if
```

Notes: Returns true if they are equal in content.

9.1.71 `itemNumberInTextList(list as NSTextListMBS, location as Integer) as Integer`

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the index of the item at the given location within the list.

Notes: list: The text list.

location: The location of the item.

Returns the index within the list.

9.1.72 `lineRangeForRange(range as NSRangeMBS) as NSRangeMBS`

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Returns the range of characters representing the line or lines containing a given range.

Notes: Range: A range within the receiver. The value must not exceed the bounds of the receiver.

The range of characters representing the line or lines containing aRange, including the line termination characters.

9.1.73 mutableCopy as NSMutableAttributedStringMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Creates a mutable copy of the attributed string.

Example:

```
// create Hello World in red
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("Hello World")
dim m as NSMutableAttributedStringMBS = a.mutableCopy

m.addAttribute(a.NSForegroundColorAttributeName, NSColorMBS.redColor, new NSRangeMBS(0, m.length))

// put it in a textarea
TextArea1.NSTextViewMBS.textStorage.setAttributedString m
```

9.1.74 NSAttachmentAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSTextAttachment

Default nil, no attachment

9.1.75 NSAuthorDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Example:

```
// get some attributed text:
dim t as new NSAttributedStringMBS
call t.initWithString("Hello World")

// set document attributes
```

```

dim dic as new Dictionary
dic.Value(t.NSAuthorDocumentAttribute) = "Test User"

// get rtf
dim rtf as string = t.RTFFromRange(dic)

// write to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim b as BinaryStream = BinaryStream.Create(f, true)
b.Write rtf

```

Notes: string containing author name.
Available in Mac OS X v10.4 and later.

9.1.76 NSBackgroundColorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
// add background color for text in text area:
```

```
TextArea1.Text = "Hello"
```

```
Dim tv As NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
Dim ts As NSTextStorageMBS = tv.TextStorage
```

```
Dim value As Color = &cFF0000
```

```
Dim offset As Integer = 0
```

```
Dim length As Integer = 5
```

```
Dim range As NSRangeMBS = NSMakeRangeMBS( offset, length )
```

```
ts.AddAttribute( NSAttributedStringMBS.NSBackgroundColorAttributeName, value, range )
```

Notes: NSColor
Default nil, no background

9.1.77 NSBackgroundColorDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSColor, representing the document-wide page background color.

Mac OS X v10.3 and earlier string constant is "BackgroundColor".

For applications linked on versions prior to Mac OS X v10.5, HTML import sets the NSBackgroundColorDocumentAttribute to NSColorMBS.whiteColor in cases in which the HTML does not specify a background color. For applications linked on Mac OS X v10.5 and later, no NSBackgroundColorDocumentAttribute is set in these cases.

Available in Mac OS X v10.4 and later.

9.1.78 NSBaselineOffsetAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing floating point value, as points offset from baseline

Default 0.0

9.1.79 NSBaseURLDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: For HTML documents; NSURL containing base URL. Previous string constant was @"BaseURL"

Available in Mac OS X v10.4 and later.

9.1.80 NSBottomMarginDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Example:

```
// get some attributed text:
dim t as new NSAttributedStringMBS
call t.initWithString("Hello World")

// set document attributes
dim dic as new Dictionary
dic.Value(t.NSBottomMarginDocumentAttribute) = 20
dic.Value(t.NSLeftMarginDocumentAttribute) = 20
```

```
dic.Value(t.NSRightMarginDocumentAttribute) = 20
dic.Value(t.NSTopMarginDocumentAttribute) = 20
dic.Value(t.NSAuthorDocumentAttribute) = "Test User"
```

```
// get rtf
dim rtf as string = t.RTFFromRange(dic)

// write to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim b as BinaryStream = BinaryStream.Create(f, true)
b.Write rtf
```

Notes: Number, containing a float, in points.
Mac OS X v10.3 and earlier string constant is "BottomMargin".
Available in Mac OS X v10.4 and later.

9.1.81 NSCategoryDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing the document's category.

Available in Mac OS X v10.6 and later.

9.1.82 NSCharacterEncodingDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing an int specifying the stringEncoding for the file; for reading and writing plain text files and writing HTML; default for plain text is the default encoding; default for HTML is UTF-8.

Mac OS X v10.3 and earlier string constant is "CharacterEncoding".

Available in Mac OS X v10.4 and later.

9.1.83 NSCharacterEncodingDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: For plain text documents; Number containing the unsigned int stringEncoding to override any encoding specified in an HTML document. Previous string constant was @"CharacterEncoding".

Available in Mac OS X v10.4 and later.

9.1.84 NSCharacterShapeAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: An integer value. The value is interpreted as Apple Type Services kCharacterShapeType selector + 1.

The default value is 0 (disable). 1 is kTraditionalCharactersSelector, and so on. Refer to <ATS/SFNTLayoutTypes.h>and "Font Features" in ATSUI Programming Guide for additional information.

Available in Mac OS X v10.0 and later.

9.1.85 NSCocoaVersionDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing a float. For RTF files only, stores the version of Cocoa with which the file was created. Absence of this value indicates RTF file not created by Cocoa or its predecessors.

Values less than 100 are pre-Mac OS X; 100 is Mac OS X v10.0 or v10.1; 102 is Mac OS X v10.2 and 10.3; values greater than 102 correspond to values of NSAppKitVersionNumber on Mac OS X v10.4 and later.

Mac OS X v10.3 and earlier string constant is "CocoaRTFVersion".

Available in Mac OS X v10.4 and later.

9.1.86 NSCommentDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing document comments.

Available in Mac OS X v10.4 and later.

9.1.87 NSCompanyDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing company or organization name.

Available in Mac OS X v10.4 and later.

9.1.88 NSConvertedDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing an int. Indicates whether the file was converted by a filter service.

If missing or 0, the file was originally in the format specified by document type. If negative, the file was originally in the format specified by document type, but the conversion to NSAttributedString may have been lossy. If 1 or more, it was converted to this type by a filter service.

Mac OS X v10.3 and earlier string constant is @"Converted".

Available in Mac OS X v10.4 and later.

9.1.89 NSCopyrightDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing document copyright info.

Available in Mac OS X v10.4 and later.

9.1.90 NSCreationTimeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSDate containing the creation date of the document; note that this is not the file system creation date of the file, but of the document.

Available in Mac OS X v10.4 and later.

9.1.91 NSCursorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSCursor

Default as returned by the NSCursor method IBeamCursor

Available in Mac OS X v10.3 and later.

9.1.92 NSDefaultAttributesDocumentAttribute as string

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

Function: One of the document attributes.

Notes: NSDictionary containing attributes to be applied to plain files. Used by reader methods. This key in options can specify the default attributes applied to the entire document contents. The document attributes can contain this key indicating the actual attributes used.

Available on Mac OS X 10.11.

9.1.93 NSDefaultAttributesDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: For plain text documents; Dictionary containing attributes to be applied to plain files. Previous string constant was @"DefaultAttributes".

Available in Mac OS X v10.4 and later.

9.1.94 NSDefaultTabIntervalDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number containing a float. Represents the document-wide default tab stop interval.

Mac OS X v10.3 and earlier string constant is "DefaultTabInterval".

Available in Mac OS X v10.4 and later.

9.1.95 NSDocFormatTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Microsoft Word document.

9.1.96 NSDocumentTypeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: How the document was interpreted; one of the values in "Document Types."

Mac OS X v10.3 and earlier string constant is "DocumentType".

Available in Mac OS X v10.4 and later.

9.1.97 NSDocumentTypeDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: One of the document types described in "Document Types," indicating a document type to be forced when loading the document. Previous string constant was @"DocumentType".

Available in Mac OS X v10.4 and later.

9.1.98 NSEditorDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing name of person who last edited the document.

Available in Mac OS X v10.4 and later.

9.1.99 NSExcludedElementsDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString related to HTML generation.

Notes: An Array object containing string objects, representing HTML elements not to be used in generated HTML.

Available in Mac OS X v10.4 and later.

NSExcludedElementsDocumentAttribute allows control over the tags used. The recognized values in the NSExcludedElementsDocumentAttribute array are (case-insensitive) HTML tags, plus DOCTYPE (representing a doctype declaration) and XML (representing an XML declaration). By default, if this attribute is not present, the excluded elements will be those deprecated in HTML 4 (APPLET, BASEFONT, CENTER, DIR, FONT, ISINDEX, MENU, S, STRIKE, and U) plus XML. If XML is on the list, HTML forms are used; if XML is not on the list, XHTML forms are used where there is a distinction.

9.1.100 NSExpansionAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing floating point value, as log of expansion factor to be applied to glyphs

Default 0.0, no expansion

Available in Mac OS X v10.3 and later.

9.1.101 `NSFileTypeDocumentAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a `NSAttributedString`.

Notes: string indicating which document type was used to interpret the document, specified as a UTI; for reading, this is available along with `NSDocumentTypeDocumentAttribute`, but for writing the two are mutually exclusive.

Available in Mac OS X v10.6 and later.

9.1.102 `NSFileTypeDocumentOption` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a `NSAttributedString`.

Notes: string indicating a document type to be forced when loading the document, specified as a UTI string; mutually exclusive with `NSDocumentTypeDocumentOption`.

Available in Mac OS X v10.6 and later.

9.1.103 NSFontAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
MsgBox NSAttributedStringMBS.NSFontAttributeName
```

```
Dim NSGraphics as New NSGraphicsMBS()
Dim NSFont as NSFontMBS = NSFontMBS.fontWithName("Helvetica", 11.0)
Dim NSAttributes as New Dictionary
```

```
NSAttributes.value(NSAttributedStringMBS.NSFontAttributeName) = nsfont
```

```
Dim stringWidth as Double = NSGraphics.sizeWithAttributes("Hello World", NSAttributes).Width
MsgBox("StringWidth from NSGraphicsMBS: " + Str(stringWidth))
```

Notes: NSFont

Default Helvetica 12-point

9.1.104 NSForegroundColorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
TextArea1.Text = "Hello"
```

```
Dim tv As NSTextViewMBS = TextArea1.NSTextViewMBS
Dim ts As NSTextStorageMBS = tv.TextStorage
```

```
Dim value As Color = &cFF0000
```

```
Dim offset As Integer = 0
```

```
Dim length As Integer = 5
```

```
Dim range As NSRangeMBS = NSMakeRangeMBS( offset, length )
ts.AddAttribute( NSAttributedStringMBS.NSForegroundColorAttributeName, value, range )
```

Notes: NSColor

Default blackColor

Available in Mac OS X v10.0 and later.

9.1.105 NSGlyphInfoAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: The name of an NSGlyphInfo object.

NSLayoutManager assigns the glyph specified by this glyph info to the entire attribute range, provided that its contents match the specified base string, and that the specified glyph is available in the font specified by NSFontAttributeName.

Available in Mac OS X v10.2 and later.

9.1.106 NSHTMLTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Hypertext Markup Language (HTML) document.

9.1.107 NSHyphenationFactorDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing a float; 0 = off, 1 = full hyphenation.

Mac OS X v10.3 and earlier string constant is "HyphenationFactor".

Available in Mac OS X v10.4 and later.

9.1.108 NSKernAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing floating point value, as points by which to modify default kerning

Default nil, use default kerning specified in font file; 0.0, kerning off; non-zero, points by which to modify default kerning

Available in Mac OS X v10.0 and later.

9.1.109 NSKeywordsDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Array of string, containing keywords.

Available in Mac OS X v10.4 and later.

9.1.110 NSLeftMarginDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing a float, in points.

Mac OS X v10.3 and earlier string constant is "LeftMargin".

Available in Mac OS X v10.4 and later.

9.1.111 NSLigatureAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing integer

Default 1, standard ligatures; 0, no ligatures; 2, all ligatures.

9.1.112 NSLinkAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSURL (preferred) or string

Default nil, no link

Available in Mac OS X v10.0 and later.

9.1.113 NSMacSimpleTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Macintosh SimpleText document.

9.1.114 NSManagerDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing the name of the author's manager.

Available in Mac OS X v10.6 and later.

9.1.115 NSMarkedClauseSegmentAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing an integer, as an index in marked text indicating clause segments

Available in Mac OS X v10.5 and later.

9.1.116 NSModificationTimeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSDate containing the modification date of the document contents.

Available in Mac OS X v10.4 and later.

9.1.117 NSObliquenessAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing floating point value, as skew to be applied to glyphs

Default 0.0, no skew

Available in Mac OS X v10.3 and later.

9.1.118 NSOfficeOpenXMLTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: ECMA Office Open XML text document format.

Available in Mac OS X v10.5 and later.

9.1.119 NSOpenDocumentTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: OASIS Open Document text document format.
Available in Mac OS X v10.5 and later.

9.1.120 NSPaperSizeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSValue, containing NSSize.
Mac OS X v10.3 and earlier string constant is "PaperSize".
Available in Mac OS X v10.4 and later.

9.1.121 NSParagraphStyleAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

// change line spacing in a Label

```
dim n as NSTextFieldMBS = Label1.NSTextFieldMBS
dim a as NSAttributedStringMBS = n.attributedStringValue

dim p as NSParagraphStyleMBS = a.attributeAtIndex(a.NSParagraphStyleAttributeName, 0)
dim m as NSMutableParagraphStyleMBS = p.mutableCopy
m.setLineSpacing 5

dim s as NSMutableAttributedStringMBS = a.mutableCopy
s.addAttribute(a.NSParagraphStyleAttributeName, m, new NSRangeMBS(0, s.length))

n.attributedStringValue = s
```

Notes: NSParagraphStyle

Default as returned by the NSParagraphStyle method defaultParagraphStyle

9.1.122 `NSPlainTextDocumentType` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the `NSDocumentTypeDocumentAttribute` key in the document attributes dictionary.

Notes: Plain text document.

9.1.123 `NSPrefixSpacesDocumentAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for `NSAttributedString` related to HTML generation.

Notes: An `NSNumber` containing an integer (default 0) representing the number of spaces per level by which to indent certain nested HTML elements.

Available in Mac OS X v10.4 and later.

`NSPrefixSpacesDocumentAttribute` allows some control over formatting.

9.1.124 `NSReadOnlyDocumentAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a `NSAttributedString`.

Notes: Number, containing int. If missing or 0 or negative, not read only; 1 or more, read only.

Note that this has nothing to do with the file system protection on the file, but instead can affect how the file should be displayed to the user.

Mac OS X v10.3 and earlier string constant is `"ReadOnly"`.

Available in Mac OS X v10.4 and later.

9.1.125 `NSRightMarginDocumentAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a `NSAttributedString`.

Notes: Number, containing a float, in points.

Mac OS X v10.3 and earlier string constant is `"RightMargin"`.

Available in Mac OS X v10.4 and later.

9.1.126 NSRTFDTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Rich text format with attachments document.

9.1.127 NSRTFTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Rich text format document.

9.1.128 NSShadowAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSShadow

Default nil, no shadow.

Available in Mac OS X v10.3 and later.

9.1.129 NSSpellingStateAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: An integer value. Defaults to 0, indicating no grammar or spelling error. See "NSSpellingStateAttributeName Flags" for possible values.

This key is available in Mac OS X v10.2 and later, but its interpretation changed in Mac OS X v10.5. Previously, any non-zero value caused the spelling indicator to be displayed. For Mac OS X v10.5 and later, the (integer) value is treated as being composed of the spelling and grammar flags.

Available in Mac OS X v10.5 and later.

9.1.130 NSStrikethroughColorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSColor

Default nil, same as foreground color

Available in Mac OS X v10.3 and later.

9.1.131 NSStrikethroughStyleAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
// underline selected text in TextArea1

dim t as NSTextViewMBS = TextArea1.NSTextViewMBS
dim s as NSTextStorageMBS = t.textStorage

const NSUnderlineStyleSingle = 1

dim d as Dictionary = t.selectedTextAttributes
d.Value(NSAttributedStringMBS.NSStrikethroughStyleAttributeName) = NSUnderlineStyleSingle
t.selectedTextAttributes = d
```

Notes: Number containing integer

Default 0, no strikethrough. See "Underlining Patterns", "Underlining Styles", and "Underline Masks" in Apple Documentation for mask values.

Available in Mac OS X v10.3 and later.

9.1.132 NSStrokeColorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSColor

Default nil, same as foreground color

Available in Mac OS X v10.3 and later.

9.1.133 NSStrokeWidthAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
TextArea1.Text = "Hello World!"
```

```
Dim v As NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
Dim a As NSTextStorageMBS = v.textStorage
```

```
a.addAttribute(a.NSStrokeWidthAttributeName, 2.0, New NSRangeMBS(6,5))
```

Notes: Number containing floating point value, as percent of font point size

Default 0, no stroke; positive, stroke alone; negative, stroke and fill (a typical value for outlined text would be 3.0)

Available in Mac OS X v10.3 and later.

9.1.134 NSSubjectDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing subject of document.

Available in Mac OS X v10.4 and later.

9.1.135 NSSuperscriptAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing integer

Default 0

9.1.136 NSTextAlternativesAttributeName as string

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

Function: One of the attribute names.

Notes: An NSTextAlternatives object. Used primarily as a temporary attribute, with primaryString equal to the substring for the range to which it is attached, and alternativeStrings representing alternatives for that string that may be presented to the user.

Available on OS X 10.8 and newer.

9.1.137 `NSTextEffectAttributeName` as string

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

Function: One of the attribute names.

Notes: NSString, default nil: no text effect

9.1.138 `NSTextEffectLetterpressStyle` as string

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

Function: One of the text effect names.

Notes: Available on Mac OS X 10.10 or newer.

9.1.139 `NSTextEncodingNameDocumentAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString related to HTML generation.

Notes: An string object containing the name, IANA or otherwise, of a text encoding to be used; mutually exclusive with NSCharacterEncodingDocumentAttribute.

Available in Mac OS X v10.4 and later.

Either NSCharacterEncodingDocumentAttribute or NSTextEncodingNameDocumentAttribute may be used to control the encoding used for generated HTML; character entities are used for characters not representable in the specified encoding.

9.1.140 `NSTextEncodingNameDocumentOption` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: string containing the name, IANA or otherwise, of a text encoding to override any encoding specified in an HTML document. Mutually exclusive with @"CharacterEncoding". Previous string constant was @"TextEncodingName".

Available in Mac OS X v10.4 and later.

9.1.141 `NSTextLayoutSectionOrientation` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the constants used as by the `NSLayoutOrientationSectionsAttribute`.

Notes: An `NSNumber` containing an `NSTextLayoutOrientation` value. The default value is `NSTextLayoutOrientationHorizontal`.

Available in Mac OS X v10.7 and later.

9.1.142 `NSTextLayoutSectionRange` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the constants used as by the `NSLayoutOrientationSectionsAttribute`.

Notes: An `NSValue` containing an `NSRange` representing a character range. The default value is a range covering the entire string.

Available in Mac OS X v10.7 and later.

9.1.143 `NSTextLayoutSectionsAttribute` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for `NSAttributedString`.

Notes: An `Array` containing dictionaries. Each dictionary describes a layout orientation section. The dictionary can have two attributes: `NSTextLayoutSectionOrientation` and `NSTextLayoutSectionRange`. When there is a gap between sections, it's assumed to have `NSTextLayoutOrientationHorizontal`.

Available in Mac OS X v10.7 and later.

9.1.144 `NSTextSizeMultiplierDocumentOption` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for `NSAttributedString`.

Notes: Specifies a scale factor for font sizes.. Number containing float, default 1.0; for HTML only, corresponding to `WebView`'s `textSizeMultiplier`.

There is no textual equivalent for Mac OS X v10.3.

Available in Mac OS X v10.4 and later.

9.1.145 `NSTimeoutDocumentOption` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for `NSAttributedString`.

Notes: Number containing float. Time in seconds to wait for a document to finish loading. Previous string constant was `@"Timeout"`.

Available in Mac OS X v10.4 and later.

9.1.146 NSTitleDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: string containing document title.

Available in Mac OS X v10.4 and later.

9.1.147 NSToolTipAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: string

Default nil, no tooltip

Available in Mac OS X v10.3 and later.

9.1.148 NSTopMarginDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Number, containing a float, in points.

Mac OS X v10.3 and earlier string constant is "TopMargin".

Available in Mac OS X v10.4 and later.

9.1.149 NSUnderlineColorAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: NSColor

Default nil, same as foreground color

Available in Mac OS X v10.3 and later.

9.1.150 NSUnderlineStyleAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: Number containing integer

Default 0, no underline. See "Underlining Patterns", "Underlining Styles", and "Underline Masks" in Apple Documentation for mask values.

9.1.151 NSVerticalGlyphFormAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: An Number containing an integer value, 0 means horizontal text and 1 indicates vertical text. If no value specified, it's interpreted to determine the setting based on higher-level vertical orientation settings such as NSTextLayoutOrientation . The behavior for any other value is undefined.

Available in Mac OS X v10.7 and later.

9.1.152 NSViewModeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSValue, containing an int; 0 = normal; 1 = page layout (use value of "PaperSize").

Mac OS X v10.3 and earlier string constant is "ViewMode".

Available in Mac OS X v10.4 and later.

9.1.153 NSViewSizeModeDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: NSValue, containing NSSize.

Mac OS X v10.3 and earlier string constant is "ViewSize".

Available in Mac OS X v10.4 and later.

9.1.154 NSViewZoomDocumentAttribute as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for document wide attributes in a NSAttributedString.

Notes: Mac OS X v10.3 and earlier string constant is "ViewZoom".

NSNumber, containing a float; 100 = 100% zoom.

Available in Mac OS X v10.4 and later.

9.1.155 NSWebArchiveTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Web Kit WebArchive document.

9.1.156 NSWebPreferencesDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: WebPreferences; for HTML only, specifies a WebPreferences object. If not present, a default set of preferences is used. Previous string constant was @"WebPreferences".

Available in Mac OS X v10.4 and later.

9.1.157 NSWebResourceLoadDelegateDocumentOption as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Notes: WebResourceLoadDelegateMBS; for HTML only, specifies an object to serve as the web resource loading delegate.

If not present, a default delegate is used that permits the loading of subsidiary resources but does not respond to authentication challenges. Previous string constant was @"WebResourceLoadDelegate".

Available in Mac OS X v10.4 and later.

9.1.158 NSWordMLTextDocumentType as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the values for the NSDocumentTypeDocumentAttribute key in the document attributes dictionary.

Notes: Microsoft Word XML (WordML schema) document.

9.1.159 NSWritingDirectionAttributeName as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the attribute names for NSAttributedString.

Example:

```
const NSWritingDirectionNatural = -1 // Determines direction using the Unicode Bidi Algorithm rules P2
and P3
const NSWritingDirectionLeftToRight = 0 // Left to right writing direction
const NSWritingDirectionRightToLeft = 1 // Right to left writing direction

const NSTextWritingDirectionEmbedding = 0
const NSTextWritingDirectionOverride = 2

dim t as NSTextStorageMBS = TextArea1.NSTextViewMBS.textStorage

// get hello in arabic
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("ÿÖÿ±ÿÿ@ÿß")
dim m as NSMutableAttributedStringMBS = a.mutableCopy

// now set attributes for right to left
m.addAttribute t.NSWritingDirectionAttributeName, array(NSWritingDirectionRightToLeft+NSTextWritingDirectionOverride), new NSRangeMBS(0,m.Length)

// and add to textarea
t.appendAttributedString m
```

Notes: An Array of Numbers.

This provides a means to override the default bidi algorithm, equivalent to the use of bidi control characters LRE, RLE, LRO, or RLO paired with PDF, as a higher-level attribute. This is the NSAttributedString equivalent of HTML's dir attribute and/or BDO element. The array represents nested embeddings or overrides, in order from outermost to innermost. The values of the Numbers should be 0, 1, 2, or 3, for LRE, RLE, LRO, or RLO respectively; these should be regarded as NSWritingDirectionLeftToRight or NSWritingDirectionRightToLeft plus NSTextWritingDirectionEmbedding or NSTextWritingDirectionOverride. Available in Mac OS X v10.6 and later.

9.1.160 paragraphRangeForRange(range as NSRangeMBS) as NSRangeMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Returns the range of characters representing the paragraph or paragraphs containing a given range.

Notes: Range: A range within the receiver. The range must not exceed the bounds of the receiver.

The range of characters representing the paragraph or paragraphs containing range, including the paragraph termination characters.

A paragraph is any segment of text delimited by a carriage return (U+000D), newline (U+000A), or paragraph separator (U+2029).

9.1.161 rangeOfTextBlock(textBlock as NSTextBlockMBS, location as Integer) as NSRangeMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the range of the individual text block that contains the specified location.

9.1.162 rangeOfTextList(list as NSTextListMBS, location as Integer) as NSRangeMBS

Plugin Version: 18.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the range of the given text list that contains the given location.

Notes: list: The text list.

location: The location in the text list.

Returns the range of the given text list containing the location.

9.1.163 rangeOfTextTable(textTable as NSTextTableMBS, location as Integer) as NSRangeMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the range of the specified text table that contains the specified location.

9.1.164 rtf as MemoryBlock

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Returns the content of this attributed string as a RTF string.

Example:

```
dim s as NSAttributedStringMBS
```

```
s=new NSAttributedStringMBS
```

```

if s.initWithString("Hello") then
MsgBox s.RTF
end if

// shows this text:
// { \rtf1\ansi\ansicpg1252\cocoartf949
// { \fonttbl\f0\fswiss\fcharset0 Helvetica; }
// { \colortbl;\red255\green255\blue255; }
// \pard\tx560\tx1120\tx1680\tx2240\tx2800\tx3360\tx3920\tx4480\tx5040\tx5600\tx6160\tx6720\ql\qnat-
// \pardirnatural
//
// \f0\fs24 \cf0 Hello }

```

Notes: Same as RTFFromRange(0,length)

9.1.165 RTFDFileWrapperFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as NSFileWrapperMBS

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an NSFileWrapper object that contains an RTFD document corresponding to the characters and attributes within the given range.

Notes: offset and length: The range.

documentAttributes: A required dictionary specifying the document attributes. The dictionary contains values from Document Types and must at least contain NSDocumentTypeDocumentAttribute. If there are no document-level attributes, dict can be nil.

Returns a file wrapper containing the RTFD data.

The file wrapper also includes the document-level attributes in docAttributes, as explained in RTF Files and Attributed Strings.

Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Äôs characters.

You can save the file wrapper using the NSFileWrapper method writeToFile.

9.1.166 RTFDFromRange(documentAttributes as dictionary = nil) as MemoryBlock

Plugin Version: 9.4, Platform: macOS, Targets: All.

Function: Creates RTFD data from the current string range.

Notes: Returns nil on failure.

Same as `RTFDFromRange(0,length)`

`documentAttributes` can optionally be a dictionary with document attributes like author or title.

See also:

- 9.1.167 `RTFDFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock` 980

9.1.167 `RTFDFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates RTFD data from the current string range.

Notes: Returns nil on failure.

`documentAttributes` can optionally be a dictionary with document attributes like author or title.

See also:

- 9.1.166 `RTFDFromRange(documentAttributes as dictionary = nil) as MemoryBlock` 979

9.1.168 `RTFFFromRange(documentAttributes as dictionary = nil) as MemoryBlock`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates RTF data from whole string.

Notes: Returns nil on failure.

Same as `RTFFFromRange(0,length)`

`documentAttributes` can optionally be a dictionary with document attributes like author or title.

See also:

- 9.1.169 `RTFFFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock` 980

9.1.169 `RTFFFromRange(offset as Integer, length as Integer, documentAttributes as dictionary = nil) as MemoryBlock`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Creates RTF data from the current string range.

Example:

```
// get some attributed text:
dim t as new NSAttributedStringMBS
call t.initWithString("Hello World")
```

```

// set document attributes
dim dic as new Dictionary
dic.Value(t.NSBottomMarginDocumentAttribute) = 20
dic.Value(t.NSLeftMarginDocumentAttribute) = 20
dic.Value(t.NSRightMarginDocumentAttribute) = 20
dic.Value(t.NSTopMarginDocumentAttribute) = 20
dic.Value(t.NSAuthorDocumentAttribute) = "Test User"

// get rtf
dim rtf as string = t.RTFFromRange(dic)

// write to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
dim b as BinaryStream = BinaryStream.Create(f, true)
b.Write rtf

```

Notes: Returns nil on failure.

documentAttributes can optionally be a dictionary with document attributes like author or title. See also:

- 9.1.168 RTFFromRange(documentAttributes as dictionary = nil) as MemoryBlock

9.1.170 Properties

9.1.171 containsAttachments as boolean

Plugin Version: 9.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether this attributed string has attachments.

Notes: (Read only property)

9.1.172 Handle as Integer

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: Must not be nil to have the object being valid.
(Read and Write property)

9.1.173 length as Integer

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Length of text in characters.

Example:

```
dim n as NSAttributedStringMBS

n=new NSAttributedStringMBS
if n.initWithString("test") then
  MsgBox str(n.length) // shows 4
else
  MsgBox "failed"
end if
```

Notes: (Read only property)

9.1.174 Range as NSRangeMBS

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Creates range for whole text.

Notes: (Read only property)

9.1.175 text as string

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: The plain text.

Notes: (Read only property)

9.1.176 Constants

Constants

Constant	Value	Description
NSSpellingStateSpellingFlag	0	One of the constants for spelling state. Flag for spelling issues.

Spelling Constants.

Constant	Value	Description
NSSpellingStateGrammarFlag	1	Flag for grammar issues.

Underline Constants.

Constant	Value	Description
NSUnderlineByWord	&h8000	Draw the line only underneath or through words, not whitespace.
NSUnderlinePatternDash	&h0200	Draw a line of dashes.
NSUnderlinePatternDashDot	&h0300	Draw a line of alternating dashes and dots.
NSUnderlinePatternDashDotDot	&h0400	Draw a line of alternating dashes and two dots.
NSUnderlinePatternDot	&h0100	Draw a line of dots.
NSUnderlinePatternSolid	0	Draw a solid line.
NSUnderlineStyleDouble	9	Draw a double line.
NSUnderlineStyleNone	0	Do not draw a line.
NSUnderlineStyleSingle	1	Draw a single line.
NSUnderlineStyleThick	2	Draw a thick line.

Writing Direction Constants.

Constant	Value	Description
NSWritingDirectionEmbedding	0	Text is embedded in text with another writing direction. For example, an English quotation in the middle of an Arabic sentence could be marked as being embedded left-to-right text.
NSWritingDirectionOverride	1	Enables character types with inherent directionality to be overridden when required for special cases, such as for part numbers made of mixed English, digits, and Hebrew letters to be written from right to left.

9.2 class NSFileWrapperMBS

9.2.1 class NSFileWrapperMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The NSFileWrapper class provides access to the attributes and contents of file-system nodes.

Example:

```
// insert a file to textview

Public Sub InsertFile(textview as NSTextViewMBS, f as FolderItem)
// read to file
dim b as BinaryStream = BinaryStream.Open(f)
dim s as string = b.Read(b.Length)

// build wrapper
dim fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
dim fileAttachment as new NSTextAttachmentMBS(fileWrapper)
dim attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: A file-system node is a file, directory, or symbolic link. Instances of this class are known as file wrappers.

File wrappers represent a file-system node as an object that can be displayed as an image (and possibly edited in place), saved to the file system, or transmitted to another application.

There are three types of file wrappers:

- Regular-file file wrapper: Represents a regular file.
- Directory file wrapper: Represents a directory.
- Symbolic-link file wrapper: Represents a symbolic link.

A file wrapper has these attributes:

- **Filename.** Name of the file-system node the file wrapper represents.
- **file-system attributes.** See `NSFileManager` Class Reference for information on the contents of the attributes dictionary.
- **Regular-file contents.** Applicable only to regular-file file wrappers.
- **File wrappers.** Applicable only to directory file wrappers.
- **Destination node.** Applicable only to symbolic-link file wrappers.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 16.0pr6](#)
- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.0](#)
- [MBS Xojo / Real Studio Plugins, version 14.5pr4](#)

9.2.2 Methods

9.2.3 `addFileWrapper(child as NSFileWrapperMBS) as String`

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

Function: Adds a child file wrapper to the receiver, which must be a directory file wrapper.

Notes: child: File wrapper to add to the directory.

Returns Dictionary key used to store fileWrapper in the directory,Äôs list of file wrappers. The dictionary key is a unique filename, which is the same as the passed-in file wrapper's preferred filename unless that name is already in use as a key in the directory,Äôs dictionary of children. See Working with Directory Wrappers in File System Programming Guide for more information about the file-wrapper list structure.

Use this method to add an existing file wrapper as a child of a directory file wrapper. If the file wrapper does not have a preferred filename, set the preferredFilename property to give it one before calling addFileWrapper. To create a new file wrapper and add it to a directory, use the addRegularFileWithContents method.

Special Considerations

This method raises `NSInternalInconsistencyException` if the receiver is not a directory file wrapper.

This method raises `NSInvalidArgumentException` if the child file wrapper doesn,Äôt have a preferred name.

9.2.4 `addRegularFileWithContents(Data as MemoryBlock, preferredFilename as string) as String`

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

Function: Creates a regular-file file wrapper with the given contents and adds it to the receiver, which must be a directory file wrapper.

Notes: data: Contents for the new regular-file file wrapper.

filename: Preferred filename for the new regular-file file wrapper.

Returns Dictionary key used to store the new file wrapper in the directory,Ãs list of file wrappers. The dictionary key is a unique filename, which is the same as the passed-in file wrapper’s preferred filename unless that name is already in use as a key in the directory’s dictionary of children. See Working with Directory Wrappers in File System Programming Guide for more information about the file-wrapper list structure.

This is a convenience method. The default implementation allocates a new file wrapper, initializes it with `initRegularFileWithContents`, set its `preferredFilename` property, adds it to the directory with `addFileWrapper`, and returns what `addFileWrapper` returned.

This method raises `NSInternalInconsistencyException` if the receiver is not a directory file wrapper.

This method raises `NSInvalidArgumentException` if you pass `nil` or an empty value for filename.

Available in OS X v10.0 and later.

9.2.5 Constructor

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The constructor.

9.2.6 `initWithFileWrappers(childrenByPreferredName as Dictionary) as NSFileWrapperMBS`

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

Function: Initializes the receiver as a directory file wrapper, with a given file-wrapper list.

Notes: childrenByPreferredName: Key-value dictionary of file wrappers with which to initialize the receiver. The dictionary must contain entries whose values are the file wrappers that are to become children and whose keys are filenames. See Working with Directory Wrappers in File System Programming Guide for more information about the file-wrapper list structure.

Returns initialized file wrapper for fileWrappers.

After initialization, the file wrapper is not associated with a file-system node until you save it using `writeToURL:options:originalContentsURL:`.

The receiver is initialized with open permissions: anyone can read, write, or modify the directory on disk.

If any file wrapper in the directory doesn't have a preferred filename, its preferred name is automatically set to its corresponding key in the `childrenByPreferredName` dictionary.

9.2.7 `initWithRegularFileWithContents(data as MemoryBlock) as NSFileWrapperMBS`

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes the receiver as a regular-file file wrapper.

Example:

```
dim content as MemoryBlock = "Hello World"
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(content)
f.filename = "HelloWorld.txt"
```

Notes: After initialization, the file wrapper is not associated with a file-system node until you save it using `writeToURL`.

The file wrapper is initialized with open permissions: anyone can write to or read the file wrapper.

9.2.8 `initWithFile(File as folderItem, Options as Integer, byref error as NSErrorMBS) as NSFileWrapperMBS`

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes a file wrapper instance whose kind is determined by the type of file-system node located by the folderitem.

Example:

```
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)

if e <> nil then
  MsgBox e.localizedDescription
else
  dim dic as Dictionary = f.fileAttributes
  Break // read dic in debugger
```

end if

Notes: File: FolderItem of the file-system node the file wrapper is to represent.

options: Option flags for reading the node located at folderitem. Can be NSFileWrapperReadingImmediate and/or NSFileWrapperReadingWithoutMapping.

Error: If an error occurs, upon return contains an NSErrorMBS object that describes the problem.

Returns File wrapper for the file-system node at folderitem. May be a directory, file, or symbolic link, depending on what is located at the folderitem. Returns nil if reading is not successful.

If folderitem is a directory, this method recursively creates file wrappers for each node within that directory. Use the fileWrappers property to get the file wrappers of the nodes contained by the directory.

Available in OS X v10.6 and later.

9.2.9 initWithSerializedRepresentation(data as MemoryBlock) as NSFileWrapperMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes the receiver as a regular-file file wrapper from given serialized data.

Example:

```
// get a wrapper
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)
dim data as MemoryBlock = f.serializedRepresentation

// later restore it
dim other as NSFileWrapperMBS = NSFileWrapperMBS.initWithSerializedRepresentation(data)
MsgBox other.preferredFilename
```

Notes: serializedRepresentation: Serialized representation of a file wrapper in the format used for the NSFileContentsPboardType pasteboard type. Data of this format is returned by such methods as serializedRepresentation and RTFDFromRange (in NSAttributedString).

Returns regular-file file wrapper initialized from serializedRepresentation.

The file wrapper is not associated with a file-system node until you save it using writeToURL.

9.2.10 `initWithURL(URL as string, Options as Integer, byref error as NSErrorMBS) as NSFileWrapperMBS`

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes a file wrapper instance whose kind is determined by the type of file-system node located by the URL.

Notes: url: URL of the file-system node the file wrapper is to represent.

options: Option flags for reading the node located at url. Can be `NSFileWrapperReadingImmediate` and/or `NSFileWrapperReadingWithoutMapping`.

Error: If an error occurs, upon return contains an `NSErrorMBS` object that describes the problem.

Returns File wrapper for the file-system node at url. May be a directory, file, or symbolic link, depending on what is located at the URL. Returns nil if reading is not successful.

If url is a directory, this method recursively creates file wrappers for each node within that directory. Use the `fileWrappers` property to get the file wrappers of the nodes contained by the directory.

Available in OS X v10.6 and later.

9.2.11 `keyForFileWrapper(child as NSFileWrapperMBS) as String`

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

Function: Returns the dictionary key used by a directory to identify a given file wrapper.

Notes: child: The child file wrapper for which you want the key.

Returns Dictionary key used to store the file wrapper in the directory,Äôs list of file wrappers. The dictionary key is a unique filename, which may not be the same as the passed-in file wrapper's preferred filename if more than one file wrapper in the directory's dictionary of children has the same preferred filename. See *Working with Directory Wrappers in File System Programming Guide* for more information about the file-wrapper list structure. Returns nil if the file wrapper specified in child is not a child of the directory.

Special Considerations

This method raises `NSInternalInconsistencyException` if the receiver is not a directory file wrapper.

9.2.12 `matchesContentsOfFile(File as FolderItem) as Boolean`

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

Function: Indicates whether the contents of a file wrapper matches a directory, regular file, or symbolic link on disk.

Notes: file: FolderItem of the file-system node with which to compare the file wrapper.

Returns true when the contents of the file wrapper match the contents of url, false otherwise.

The contents of files are not compared; matching of regular files is based on file modification dates. For a directory, children are compared against the files in the directory, recursively.

Because children of directory file wrappers are not read immediately by the initWithURL:options:error: method unless the NSFileWrapperReadingImmediate reading option is used, even a newly-created directory file wrapper might not have the same contents as the directory on disk. You can use this method to determine whether the file wrapper's contents in memory need to be updated.

If the file wrapper needs updating, use the readFromURL:options:error: method with the NSFileWrapperReadingImmediate reading option.

This table describes which attributes of the file wrapper and file-system node are compared to determine whether the file wrapper matches the node on disk:

File-wrapper type	Comparison determinants
Regular file	Modification date and access permissions.
Directory	Children (recursive).
Symbolic link	Destination pathname.

Available in OS X v10.6 and later.

9.2.13 matchesContentsOfURL(URL as String) as Boolean

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

Function: Indicates whether the contents of a file wrapper matches a directory, regular file, or symbolic link on disk.

Notes: url: URL of the file-system node with which to compare the file wrapper.

Returns true when the contents of the file wrapper match the contents of url, false otherwise.

The contents of files are not compared; matching of regular files is based on file modification dates. For a directory, children are compared against the files in the directory, recursively.

Because children of directory file wrappers are not read immediately by the initWithURL:options:error:

method unless the `NSFileWrapperReadingImmediate` reading option is used, even a newly-created directory file wrapper might not have the same contents as the directory on disk. You can use this method to determine whether the file wrapper's contents in memory need to be updated.

If the file wrapper needs updating, use the `readFromURL:options:error:` method with the `NSFileWrapperReadingImmediate` reading option.

This table describes which attributes of the file wrapper and file-system node are compared to determine whether the file wrapper matches the node on disk:

File-wrapper type	Comparison determinants
Regular file	Modification date and access permissions.
Directory	Children (recursive).
Symbolic link	Destination pathname.

Available in OS X v10.6 and later.

9.2.14 `readFromFile(File as FolderItem, Options as Integer = 0, byref Error as NSErrorMBS) as Boolean`

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

Function: Recursively rereads the entire contents of a file wrapper from the specified location on disk.

Notes: File: FolderItem of the file-system node corresponding to the file wrapper.

options: Option flags for reading the node located at url.

Error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true if successful. If not successful, returns false after setting `outError` to an NSErrorMBS object that describes the reason why the file wrapper could not be reread.

When reading a directory, children are added and removed as necessary to match the file system.

Available in OS X v10.6 and later.

9.2.15 `readFromURL(URL as String, Options as Integer = 0, byref Error as NSErrorMBS) as Boolean`

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

Function: Recursively rereads the entire contents of a file wrapper from the specified location on disk.

Notes: url: URL of the file-system node corresponding to the file wrapper.

options: Option flags for reading the node located at url.

Error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true if successful. If not successful, returns false after setting outError to an NSErrorMBS object that describes the reason why the file wrapper could not be reread.

When reading a directory, children are added and removed as necessary to match the file system.
Available in OS X v10.6 and later.

9.2.16 removeFileWrapper(child as NSFileWrapperMBS)

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

Function: Removes a child file wrapper from the receiver, which must be a directory file wrapper.

Notes: This method raises NSInternalInconsistencyException if the receiver is not a directory file wrapper.

9.2.17 writeToFile(File as FolderItem, Options as Integer = 0, originalContentsURL as FolderItem = nil, byref Error as NSErrorMBS) as Boolean

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

Function: Recursively writes the entire contents of a file wrapper to a given file-system URL.

Example:

```
// get styled text from htmlviewer
dim w as WebViewMBS = HTMLViewer1.WebViewMBS
dim f as WebFrameMBS = w.mainFrame
dim v as WebFrameViewMBS = f.frameView
dim d as WebDocumentViewMBS = v.documentView
dim n as NSAttributedStringMBS = d.attributedString

// package it
dim da as new Dictionary
da.Value(n.NSDocumentTypeDocumentAttribute) = n.NSRTFDTextDocumentType

dim e as NSErrorMBS
dim fw as NSFileWrapperMBS = n.fileWrapperFromRange(0, n.Length, da, e)
if e <> nil then
  MsgBox e.LocalizedDescription
  Return
end if

// write to disk
```

```

dim file as FolderItem = SpecialFolder.Desktop.Child("test.rtf")
if fw.writeToFile(file, e) then
MsgBox "OK"
else
MsgBox e.LocalizedDescription
end if

```

Notes: File: FolderItem of the file-system node to which the file wrapper,Ãs contents are written.

options: Option flags for writing to the node located at url.

originalContentsURL: The location of a previous revision of the contents being written. The default implementation of this method attempts to avoid unnecessary I/O by writing hard links to regular files instead of actually writing out their contents when the contents have not changed. The child file wrappers must return accurate values when its filename property is accessed for this to work. Use the NSFileWrapperWritingWithNameUpdating writing option to increase the likelihood of that. Specify nil for this parameter if there is no earlier version of the contents or if you want to ensure that all the contents are written to files.

Error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true when the write operation is successful. If not successful, returns false after setting outError to an NSError object that describes the reason why the file wrapper,Ãs contents could not be written.

Available in OS X v10.6 and later.

9.2.18 writeToURL(URL as String, Options as Integer = 0, originalContentsURL as String = "", byref Error as NSErrorMBS) as Boolean

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

Function: Recursively writes the entire contents of a file wrapper to a given file-system URL.

Notes: url: URL of the file-system node to which the file wrapper,Ãs contents are written.

options: Option flags for writing to the node located at url.

originalContentsURL: The location of a previous revision of the contents being written. The default implementation of this method attempts to avoid unnecessary I/O by writing hard links to regular files instead of actually writing out their contents when the contents have not changed. The child file wrappers must return accurate values when its filename property is accessed for this to work. Use the NSFileWrapperWritingWithNameUpdating writing option to increase the likelihood of that. Specify nil for this parameter if there is no earlier version of the contents or if you want to ensure that all the contents are written to files.

Error: If an error occurs, upon return contains an NSError object that describes the problem.

Returns true when the write operation is successful. If not successful, returns false after setting outError to an NSError object that describes the reason why the file wrapper,Ãs contents could not be written.

Available in OS X v10.6 and later.

9.2.19 Properties

9.2.20 Directory as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: This property contains a boolean value indicating whether the file wrapper is a directory file wrapper. (read-only)

Notes: This property will contain true when the file wrapper is a directory file wrapper, otherwise it contains false.

Invocations of `readFromURL` may change the value of this property, if the type of the file on disk has changed. (Read only property)

9.2.21 fileAttributes as Dictionary

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: A dictionary of file attributes.

Example:

```
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)

if e <> nil then
  MsgBox e.localizedDescription
else
  dim dic as Dictionary = f.fileAttributes
  Break // read dic in debugger
end if
```

Notes: The file attributes dictionary is the same format as that returned by `attributesOfItemAtPath` (see `NSFileManagerMBS`).

(Read and Write property)

9.2.22 filename as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The filename of the file wrapper object.

Example:

```
dim content as MemoryBlock = "Hello World"
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(content)
f.filename = "HelloWorld.txt"
```

Notes: This property contains the file wrapper, the filename, or nil when the file wrapper has no corresponding file-system node.

The filename is used for record-keeping purposes only and is set automatically when the file wrapper is created from the file system using `initWithURL` and when it is saved to the file system using `writeToURL` (although this method allows you to request that the filename not be updated).

The filename is usually the same as the preferred filename, but might instead be a name derived from the preferred filename. You can use this method to get the name of a child that's just been read. Don't use this method to get the name of a child that's about to be written, because the name might be about to change; send `keyForFileWrapper` to the parent instead.
(Read and Write property)

9.2.23 fileWrappers as Dictionary

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

Function: The file wrappers contained by a directory file wrapper.

Notes: The dictionary contains entries whose values are the file wrappers and whose keys are the unique filenames that have been assigned to each one. See *Working with Directory Wrappers in File System Programming Guide* for more information about the file-wrapper list structure.

This property may contain nil if the user modifies the directory after you call `readFromURL` or `initWithURL` but before `NSFileWrapper` has read the contents of the directory. Use the `NSFileWrapperReadingImmediate` reading option to reduce the likelihood of that problem.

This property raises `NSInternalInconsistencyException` if the file wrapper object is not a directory file wrapper.
(Read only property)

9.2.24 Handle as Integer

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: Must not be 0 to have the object being valid.

(Read and Write property)

9.2.25 icon as NSImageMBS

Plugin Version: 15.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: An image that can be used to represent the file wrapper to the user.

Example:

```
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)
dim img as NSImageMBS = f.icon
dim pic as Picture = img.CopyPictureWithMask
window1.Backdrop = pic
```

Notes: An application does not have to use this icon; for example, a file viewer typically looks up icons automatically based on file extensions, and so wouldn't need this one. Similarly, if a file wrapper represents an image file, an application can display the image directly rather than a file icon.

This method may return nil if the file wrapper is a child created when its parent was read from the file system, and the child was modified before it was read. Use the `NSFileWrapperReadingImmediate` reading option to reduce the likelihood of that problem.

Because the `NSImage` object that's returned might be shared by many `NSFileWrapper` objects, you must not mutate it. If you need to mutate the returned object, make a copy first and mutate the copy instead. (Read and Write property)

9.2.26 preferredFilename as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The preferred filename for the file wrapper object.

Example:

```
// get a wrapper
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)
dim data as MemoryBlock = f.serializedRepresentation

// later restore it
dim other as NSFileWrapperMBS = NSFileWrapperMBS.initWithSerializedRepresentation(data)
```

MsgBox other.preferredFilename

Notes: This name is normally used as the dictionary key when a child file wrapper is added to a directory file wrapper. However, if another file wrapper with the same preferred name already exists in the directory file wrapper when this object is added, the filename assigned as the dictionary key may differ from the preferred filename.

When you change the preferred filename, the default implementation of this property causes the existing parent directory file wrappers to remove and re-add the child to accommodate the change. Preferred filenames of children are not preserved when you write a file wrapper to disk and then later instantiate another file wrapper by reading the file from disk. If you need to preserve the user-visible names of attachments, you have to store the names yourself.

(Read and Write property)

9.2.27 RegularFile as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: This property contains a boolean value that indicates whether the file wrapper object is a regular-file. (read-only)

Notes: This property contains true when the file wrapper object is a regular-file wrapper, otherwise it contains false. Invocations of `readFromURL` may change the value of this property if the type of the file on disk has changed.

(Read only property)

9.2.28 regularFileContents as MemoryBlock

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The contents of the file-system node associated with a regular-file file wrapper. (read-only)

Notes: This property may contain nil if the user modifies the file after you call `readFromURL` or `initWithURL` but before `NSFileWrapper` has read the contents of the file. Use the `NSFileWrapperReadingImmediate` reading option to reduce the likelihood of that problem.

This property raises `NSInternalInconsistencyException` if the file wrapper object is not a regular-file file wrapper.

(Read only property)

9.2.29 serializedRepresentation as MemoryBlock

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The contents of the file wrapper as an opaque collection of data. (read-only)

Example:

```
// get a wrapper
dim file as FolderItem = SpecialFolder.desktop.child("todo.rtf")
dim e as NSErrorMBS
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initWithFile(file, 0, e)
dim data as MemoryBlock = f.serializedRepresentation

// later restore it
dim other as NSFileWrapperMBS = NSFileWrapperMBS.initWithSerializedRepresentation(data)
MsgBox other.preferredFilename
```

Notes: This property contains an MemoryBlock in the format used by the pasteboard type NSFileContentsPboardType. This MemoryBlock is also suitable for passing to initWithSerializedRepresentation.

This property may contain nil if the user modifies the contents of the file-system node after you call readFromURL or initWithURL/File but before NSFileWrapper has read the contents of the file. Use the NSFileWrapperReadingImmediate reading option to reduce the likelihood of that problem.
(Read only property)

9.2.30 SymbolicLink as Boolean

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: A boolean that indicates whether the file wrapper object is a symbolic-link file wrapper. (read-only)

Notes: This property contains true when the file wrapper object is a symbolic-link file wrapper, false otherwise.

Invocations of readFromURL may change the value contained by this property, if the type of the file on disk has changed.

(Read only property)

9.2.31 symbolicLinkDestinationURL as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The URL referenced by the file wrapper object, which must be a symbolic-link file wrapper. (read-only)

Notes: This property may contain nil if the user modifies the symbolic link after you call readFromURL or initWithURL but before NSFileWrapper has read the contents of the link. Use the NSFileWrapperReadingImmediate reading option to reduce the likelihood of that problem.

This property raises NSInternalInconsistencyException if the file wrapper object is not a symbolic-link file

wrapper.
(Read only property)

9.2.32 Constants

Reading Options

Constant	Value	Description
<code>NSFileWrapperReadingImmediate</code>	1	If reading with this option succeeds, then subsequent uses of <code>fileWrapper.regularFileContents</code> , <code>symbolicLinkDestinationURL</code> , and <code>serializedRepresentation</code> sent to the file wrapper and all its child file wrappers will fail and return <code>nil</code> only if an actual error occurs (for example, the volume has disappeared or the file server is unreachable)—not as a result of the user moving or deleting the file. For performance reasons, <code>NSFileWrapper</code> may not read the contents of file packages immediately even when this option is chosen. For example, the contents of bundles (not all file packages are bundles) are immutable. For user, <code>NSFileWrapper</code> may read the children of such a directory lazily. You can use this option to take a snapshot of a file or folder for writing later. For example, an application like <code>TextEdit</code> can use this option when creating file wrappers to represent attachments that the user creates by copying and pasting or dragging and dropping from the Finder to a <code>TextEdit</code> document. Do not use this option when reading a document file package, because that would cause unnecessarily bad performance. For example, an application wouldn't use this option when creating file wrappers to represent attachments as it's opening a document stored in a file package. Available in OS X v10.6 and later.
<code>NSFileWrapperReadingWithoutMapping</code>	2	Whether file mapping for regular file wrappers is disallowed. You can use this option to keep <code>NSFileWrapper</code> from memory-mapping files. This is useful if you want to make sure your application doesn't hold files open (mapped files are open files), therefore preventing the user from ejecting, unmounting disk partitions, or unmounting disk images. In OS X v10.6 and later, <code>NSFileWrapper</code> memory-maps files that are on internal drives and never memory-maps files on external drives or network volumes, regardless of whether this option is used. Available in OS X v10.6 and later.

Writing Options

Constant	Value	Description
<code>NSFileWrapperWritingAtomic</code>	1	Whether writing is done atomically. You can use this option to ensure that, when overwriting a file package, overwriting either completely succeeds or completely fails, with no possibility of leaving the file package in an inconsistent state. Because this option requires additional I/O, you shouldn't use it unnecessarily. For example, don't use this option in an override of <code>- [NSDocument writeToURL]</code> , because <code>NSDocument</code> 's safe-saving is already done atomically. Available in OS X v10.6 and later.
<code>NSFileWrapperWritingWithNameUpdating</code>	2	Whether descendant file wrappers' <code>NSFileWrapper.filename</code> properties are set if the write succeeds. This option is necessary when your application passes a URL in the <code>originalContentsURL</code> parameter to the <code>writeToURL</code> method. Without this option (and reusing child file wrappers properly), subsequent invocations of <code>writeToURL</code> would not be able to reliably create hard links in a new file package, because the record of names in the old file package would be overwritten. Available in OS X v10.6 and later.

9.3 class NSMutableAttributedStringMBS

9.3.1 class NSMutableAttributedStringMBS

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: NSMutableAttributedString declares the programmatic interface to objects that manage mutable attributed strings.

Notes: You can add and remove characters (raw strings) and attributes separately or together as attributed strings. See the class description for NSAttributedString for more information about attributed strings.

Subclass of the NSAttributedStringMBS class.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 24.0](#)
- [MBS Xojo Plugins, version 24.0pr6](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4](#)
- [MBS Xojo Plugins, version 23.4pr3](#)
- [Styled Text for Labels in your Xojo iOS app](#)
- [Tip of the day: Adding links to Textarea on OS X](#)
- [Apply fonts to font PopupMenu](#)
- [Using NSTextViewMBS](#)
- [MBS Releases the MBS Real Studio plug-ins in version 12.0](#)
- [Nearly 2000 new Functions in the 9.6 prerelease of MBS](#)

Xojo Developer Magazine

- [22.2, page 9: News](#)
- [10.3, page 9: News](#)

9.3.2 Methods

9.3.3 addAttribute(name as string, value as Variant, range as NSRangeMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Adds an attribute with the given name and value to the characters in the specified range.

Example:

```
// create Hello World in red
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("Hello World")
```

```

dim m as NSMutableAttributedStringMBS = a.mutableCopy

m.addAttribute(a.NSForegroundColorAttributeName, NSColorMBS.redColor, new NSRangeMBS(0, m.length))

// put it in a textarea
TextArea1.NSTextViewMBS.textStorage.setAttributedString m

```

Notes: name: A string specifying the attribute name. Attribute keys can be supplied by another framework or can be custom ones you define. For information about where to find the system-supplied attribute keys, see the overview section in NSAttributedString Class Reference.

value: The attribute value associated with name.

Range: The range of characters to which the specified attribute/value pair applies.

You may assign any name/value pair you wish to a range of characters, in addition to the standard attributes described in the "Constants" section of NSAttributedString Additions. Raises an NSInvalidArgumentException if name or value is nil and an NSRangeException if any part of range lies beyond the end of the receiver's characters.

Plugin version 16.0 or newer ignores calls here with value = nil or empty name to avoid exceptions.

9.3.4 addAttributes(attrs as Dictionary, range as NSRangeMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Adds the given collection of attributes to the characters in the specified range.

Example:

```
TextArea1.Text = "Hello"
```

```
Dim tv As NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
Dim ts As NSTextStorageMBS = tv.TextStorage
```

```
Dim value As Color = &cFF0000
```

```
Dim offset As Integer = 0
```

```
Dim length As Integer = 5
```

```
Dim range As NSRangeMBS = NSMakeRangeMBS( offset, length )
```

```
ts.AddAttribute( NSAttributedStringMBS.NSForegroundColorAttributeName, value, range )
```

Notes: attributes: A dictionary containing the attributes to add. Attribute keys can be supplied by another framework or can be custom ones you define. For information about where to find the system-supplied

attribute keys, see the overview section in NSAttributedString Class Reference.
Range: The range of characters to which the specified attributes apply.

You may assign any name/value pair you wish to a range of characters, in addition to the standard attributes described in the "Constants" section of NSAttributedString Additions. Raises an NSInvalidArgumentException if attributes is nil and an NSRangeException if any part of range lies beyond the end of the receiver's characters.

9.3.5 appendString(attrString as NSAttributedStringMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Adds the characters and attributes of a given attributed string to the end of the receiver.

Example:

```
const NSWritingDirectionNatural = -1 // Determines direction using the Unicode Bidi Algorithm rules P2
and P3
const NSWritingDirectionLeftToRight = 0 // Left to right writing direction
const NSWritingDirectionRightToLeft = 1 // Right to left writing direction

const NSTextWritingDirectionEmbedding = 0
const NSTextWritingDirectionOverride = 2

dim t as NSTextStorageMBS = TextArea1.NSTextViewMBS.textStorage

// get hello in arabic
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("ÿŒ±ÿŸ@ÿß")
dim m as NSMutableAttributedStringMBS = a.mutableCopy

// now set attributes for right to left
m.addAttribute t.NSWritingDirectionAttributeName, array(NSWritingDirectionRightToLeft+NSTextWritingDirectionOverride), new NSRangeMBS(0,m.Length)

// and add to textarea
t.appendAttributedString m
```

Notes: attributedString: The string whose characters and attributes are added.

9.3.6 appendString(attrString as String)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Adds the text to the end of the receiver.

9.3.7 applyFontTraits(FontTraitMask as Integer, offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Applies the font attributes specified by mask to the characters in a range.

Notes: See the NSFontManager class specification for a description of the font traits available. Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Ãs characters.

9.3.8 AsCFMutableAttributedString as Variant

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new CFMutableAttributedStringMBS object pointing to same mutable attributed string.

Example:

```
// make NS version
dim n as new NSAttributedStringMBS

if n.initWithString("Hello World") then

dim m as NSMutableAttributedStringMBS = n.mutableCopy

// convert
dim c as CFMutableAttributedStringMBS = m.AsCFMutableAttributedString

// and check content
MsgBox c.String
end if
```

Notes: For passing to functions which need a CFMutableAttributedStringMBS.

9.3.9 beginEditing

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Overridden by subclasses to buffer or optimize a series of changes to the receiver's characters or attributes, until it receives a matching endEditing message, upon which it can consolidate changes and notify any observers that it has changed.

Notes: You can nest pairs of beginEditing and endEditing messages.

9.3.10 Constructor

Plugin Version: 19.0, Platform: macOS, Targets: All.

Function: The constructor to create empty mutable attributed string.

9.3.11 containsAttachmentsInRange(offset as Integer, length as Integer) as Boolean

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns true if the receiver contains a property configured (NSAttachmentAttributeName with NSAttachmentCharacter) in range.

9.3.12 deleteCharactersInRange(range as NSRangeMBS)

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Deletes the characters in the given range along with their associated attributes.

Notes: Raises an NSRangeException if any part of aRange lies beyond the end of the receiver's characters.

9.3.13 endEditing

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Overridden by subclasses to consolidate changes made since a previous beginEditing message and to notify any observers of the changes.

Notes: The NSMutableAttributedString implementation does nothing. NSTextStorage, for example, overrides this method to invoke fixAttributesInRange and to inform its NSLayoutManager objects that they need to re-lay the text.

9.3.14 fixAttachmentAttributeInRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Cleans up attachment attributes in a range, removing all attachment attributes assigned to characters other than NSAttachmentCharacter.

Notes: Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Äôs characters.

9.3.15 fixAttributesInRange(offset as Integer, length as Integer)

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

Function: Invokes the other fix... methods, allowing you to clean up an attributed string with a single message.

Notes: Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Äôs characters.

NSTextStorage subclasses that return YES from the fixesAttributesLazily method should avoid directly calling fixAttributesInRange: or else bracket such calls with beginEditing and endEditing messages.

9.3.16 fixFontAttributeInRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Fixes the font attribute in aRange, assigning default fonts to characters with illegal fonts for their scripts and otherwise correcting font attribute assignments.

Notes: For example, Kanji characters assigned a Latin font are reassigned an appropriate Kanji font. Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Äôs characters.

9.3.17 fixParagraphStyleAttributeInRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Fixes the paragraph style attributes in a range, assigning the first paragraph style attribute value in each paragraph to all characters of the paragraph.

Notes: This method extends the range as needed to cover the last paragraph partially contained. A paragraph is delimited by any of these characters, the longest possible sequence being preferred to any shorter:

- U+000D (\r or CR)
- U+000A (\n or LF)
- U+2029 (Unicode paragraph separator) \r\n, in that order (also known as CRLF)

Raises an NSRangeException if any part of aRange lies beyond the end of the receiver,Äôs characters.

9.3.18 fontAttributesInRange(offset as Integer, length as Integer) as Dictionary

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attributes which should be copied/pasted with "copy font".

9.3.19 insertAttributedString(attrString as NSAttributedStringMBS, location as UInt64)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Inserts the characters and attributes of the given attributed string into the receiver at the given index.

Notes: attributedString: The string whose characters and attributes are inserted.

index: The index at which the characters and attributes are inserted.

The new characters and attributes begin at the given index and the existing characters and attributes from the index to the end of the receiver are shifted by the length of the attributed string. Raises an NSRangeException if index lies beyond the end of the receiver's characters.

9.3.20 insertString(attrString as String, location as UInt64)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Inserts the text into the receiver at the given index.

9.3.21 removeAttribute(name as string, range as NSRangeMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Removes the named attribute from the characters in the specified range.

Notes: name: A string specifying the attribute name to remove. Attribute keys can be supplied by another framework or can be custom ones you define. For information about where to find the system-supplied attribute keys, see the overview section in NSAttributedString Class Reference.

Range: The range of characters from which the specified attribute is removed.

Raises an NSRangeException if any part of range lies beyond the end of the receiver's characters.

9.3.22 replaceCharacters(search as String, text as String, Options as Integer = 1)

Plugin Version: 24.0, Platform: macOS, Targets: All.

Function: Searches and replaces text in a mutable attributed string.

Example:

```
// make some mutable string
Dim s As New NSMutableAttributedStringMBS
```

```
s.setString "Hello World!"

// now do the replace
s.replaceCharacters("world", "everyone")

// and show result
MessageBox s.Text
```

Notes: Default options is for case insensitive search, but you can pass other options. (see search flags in PDFDocumentMBS class)

9.3.23 `replaceCharactersInRange(range as NSRangeMBS, attrString as NSAttributedStringMBS)`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Replaces the characters and attributes in a given range with the characters and attributes of the given attributed string.

Notes: Range: The range of characters and attributes replaced.

attributedString: The attributed string whose characters and attributes replace those in the specified range.

Raises an NSRangeException if any part of range lies beyond the end of the receiver's characters.

See also:

- 9.3.24 `replaceCharactersInRange(range as NSRangeMBS, text as string)` 1008

9.3.24 `replaceCharactersInRange(range as NSRangeMBS, text as string)`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Replaces the characters in the given range with the characters of the given string.

Notes: Range: A range specifying the characters to replace.

text: A string specifying the characters to replace those in range.

The new characters inherit the attributes of the first replaced character from range. Where the length of range is 0, the new characters inherit the attributes of the character preceding range if it has any, otherwise of the character following range.

Raises an NSRangeException if any part of range lies beyond the end of the receiver's characters.

See also:

- 9.3.23 `replaceCharactersInRange(range as NSRangeMBS, attrString as NSAttributedStringMBS)` 1008

9.3.25 rulerAttributesInRange(offset as Integer, length as Integer) as Dictionary

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Attributes which should be copied/pasted with "copy ruler".

9.3.26 setAlignment(alignment as Integer, offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the alignment characteristic of the paragraph style attribute for the characters in aRange to alignment.

Notes: When attribute fixing takes place, this change will affect only paragraphs whose first character was included in aRange. Raises an NSRangeException if any part of a range lies beyond the end of the receiver,Ãs characters.

9.3.27 setAttributedString(attrString as NSAttributedStringMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Replaces the receiver's entire contents with the characters and attributes of the given attributed string.

Example:

```
// create Hello World in red
dim a as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithString("Hello World")
dim m as NSMutableAttributedStringMBS = a.mutableCopy

m.addAttribute(a.NSForegroundColorAttributeName, NSColorMBS.redColor, new NSRangeMBS(0, m.length))

// put it in a textarea
TextArea1.NSTextViewMBS.textStorage.setAttributedString m
```

Notes: attributedString: The attributed string whose characters and attributes replace those in the receiver.

9.3.28 setAttributes(attrs as Dictionary, range as NSRangeMBS)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Sets the attributes for the characters in the specified range to the specified attributes.

Notes: attributes: A dictionary containing the attributes to set. Attribute keys can be supplied by another

framework or can be custom ones you define. For information about where to find the system-supplied attribute keys, see the overview section in NSAttributedString Class Reference.

Range: The range of characters whose attributes are set.

These new attributes replace any attributes previously associated with the characters in range. Raises an NSRangeException if any part of range lies beyond the end of the receiver's characters.

To set attributes for a zero-length NSMutableAttributedString displayed in a text view, use the NSTextView method setTypingAttributes.

9.3.29 setBaseWritingDirection(writingDirection as Integer, offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the base writing direction for the characters in range to writingDirection.

9.3.30 setString(attrString as String)

Plugin Version: 18.1, Platform: macOS, Targets: All.

Function: Replaces the receiver's entire contents with the given string.

9.3.31 subscriptRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Decrements the value of the superscript attribute for characters in a range by 1.

Example:

```
TextArea1.Text = "Hello World"
```

```
dim n as NSTextViewMBS = TextArea1.NSTextViewMBS
```

```
dim s as NSTextStorageMBS = n.textStorage
```

```
s.subscriptRange(6,5)
```

9.3.32 superscriptRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Increments the value of the superscript attribute for characters in a range by 1.

9.3.33 unscriptRange(offset as Integer, length as Integer)

Plugin Version: 16.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Removes the superscript attribute from the characters in a range.

9.3.34 updateAttachmentsFromPath(file as folderitem)

Plugin Version: 15.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Updates all attachments based on files contained in the RTFD file package at path.
See also:

- 9.3.35 updateAttachmentsFromPath(path as string)

1011

9.3.35 updateAttachmentsFromPath(path as string)

Plugin Version: 15.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Updates all attachments based on files contained in the RTFD file package at path.
See also:

- 9.3.34 updateAttachmentsFromPath(file as folderitem)

1011

9.4 class NSShadowMBS

9.4.1 class NSShadowMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: An NSShadow object encapsulates the attributes used to create a drop shadow during drawing operations.

Notes: Shadows are always drawn in the default user coordinate space, regardless of any transformations applied to that space. This means that rotations, translations and other transformations of the current transformation matrix (the CTM) do not affect the resulting shadow. Another way to think about this is that changes to the CTM do not move or change the apparent position of the shadow,Ã light source.

There are two positional parameters for a shadow: an x-offset and a y-offset. These values are expressed using a single size data type (CGSize on iOS, NSSize on OS X) and using the units of the default user coordinate space. Positive values for these offsets extend up and to the right.

In addition to its positional parameters, a shadow also contains a blur radius, which specifies how much a drawn object’s image mask is blurred before it is composited onto the destination. A value of 0 means there is no blur. Larger values give correspondingly larger amounts of blurring.

An NSShadow object may be used in one of two ways. First, it may be set, like a color or a font, in which case its attributes are applied to all content drawn thereafter—or at least until another shadow is applied or a previous graphics state is restored. Second, it may be used as the value for the NSShadowAttributeName text attribute, in which case it is applied to the glyphs corresponding to the characters bearing this attribute.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 16.2pr3](#)
- [MBS Xojo / Real Studio Plugins, version 16.1pr1](#)

9.4.2 Methods

9.4.3 Constructor

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The constructor.

Notes: Initialized with 0 as its offset, 0 as its blur radius, and the default color as its color.

9.4.4 copy as NSShadowMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Creates a copy of the shadow object.

9.4.5 set

Plugin Version: 16.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the shadow of subsequent drawing operations to the shadow represented by the receiver.

Notes: The shadow attributes of the receiver are used until another shadow is set or until the graphics state is restored.

9.4.6 Properties

9.4.7 Handle as Integer

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

9.4.8 shadowBlurRadius as Double

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The blur radius of the shadow.

Notes: This property contains the blur radius, as measured in the default user coordinate space. A value of 0 indicates no blur, while larger values produce correspondingly larger blurring. The default value is 0. (Read and Write property)

9.4.9 shadowColor as NSColorMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The color of the shadow.

Notes: The default shadow color is black with an alpha of 1/3. If you set this property to nil, the shadow is not drawn.

(Read and Write property)

9.4.10 shadowOffset as NSSizeMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The offset values of the shadow.

Notes: This property contains the horizontal and vertical offset values, specified using the width and height fields of the NSSize data type. These offsets are measured using the default user coordinate space and are not affected by custom transformations. This means that positive values always extend down and to the right from the user's perspective.

(Read and Write property)

9.5 class NSTextAttachmentMBS

9.5.1 class NSTextAttachmentMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: An attachment to a text.

Notes: NSTextAttachment objects are used by the NSAttributedString class cluster as the values for attachment attributes (stored in the attributed string under the key named NSAttachmentAttributeName). The objects you create with this class are referred to as text attachment objects, or when no confusion will result, as text attachments or merely attachments.

A text attachment object contains an NSFileWrapper in NSFileWrapper Class Reference object, which in turn holds the contents of the attached file. It also uses a cell object conforming to the NSTextAttachmentCell in NSTextAttachmentCell Protocol Reference protocol to draw and handle mouse events. Most of the behavior of a text attachment is relegated to the file wrapper and the attachment cell. See the corresponding class and protocol specifications for more information.

See the NSAttributedString in NSAttributedString Class Reference and NSTextView in NSTextView Class Reference class specifications for general information on text attachments.

see also

https://developer.apple.com/library/mac/documentation/Cocoa/Reference/ApplicationKit/Classes/NSTextAttachment_Class/index.html#//apple_ref/occ/instm/NSTextAttachment/initWithFileWrapper:

Blog Entries

- [RTF functions in MBS Plugins](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr4](#)
- [MonkeyBread Software Releases the MBS Xojo / Real Studio plug-ins in version 15.0](#)
- [MBS Xojo / Real Studio Plugins, version 14.5pr4](#)

9.5.2 Methods

9.5.3 attributedStringWithAttachment(attachment as NSTextAttachmentMBS) as NSAttributedStringMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Creates an attributed string with an attachment.

Example:

```
dim content as MemoryBlock = "Hello World"  
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(content)
```

```
f.filename = "HelloWorld.txt"
```

```
dim a as new NSTextAttachmentMBS(f)
dim s as NSAttributedStringMBS = NSTextAttachmentMBS.attributedStringWithAttachment(a)
```

Notes: This is a convenience method for creating an attributed string containing an attachment using `NSAttachmentCharacter` as the base character.

9.5.4 Constructor(fileWrapper as NSFileWrapperMBS)

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Initializes a newly allocated `NSTextAttachment` object to contain the given file wrapper.

Example:

```
dim content as MemoryBlock = "Hello World"
dim f as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(content)
f.filename = "HelloWorld.txt"
```

```
dim a as new NSTextAttachmentMBS(f)
```

```
Break // inspect in debugger
```

Notes: If `fileWrapper` contains an image file that the receiver can interpret as an `NSImage` object, sets the attachment cell's image to the `NSImage` rather than to the icon of `fileWrapper`.

See also:

- 9.5.5 Constructor(image as NSImageMBS)

1016

9.5.5 Constructor(image as NSImageMBS)

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Initializes a newly allocated `NSTextAttachment` object with a cell and given image.

Example:

```
Dim pic As Picture = LogoMBS(500)
Dim image As New NSImageMBS(pic)
```

```
// build attachment
```

```
Dim imageAttachment As New NSTextAttachmentMBS(image)
```

```
Dim attributedString As NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(imageAttachment)
```

```
// insert image  
Dim textView As NSTextViewMBS = TextArea1.NSTextViewMBS  
textView.insertText attributedString
```

See also:

- 9.5.4 Constructor(fileWrapper as NSFileWrapperMBS)

1016

9.5.6 Properties

9.5.7 attachmentCell as Variant

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The attachment cell.

Notes: As our plugin currently has no NSTextAttachmentCell class, we just give you a NSCell object. This allows accessing most of the properties.

Value is a NSCellMBS declared as Variant to reduce plugin dependencies.
(Read only property)

9.5.8 fileWrapper as NSFileWrapperMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The file wrapper.

Notes: The file wrapper holds the contents of the attached file.
(Read and Write property)

9.5.9 Handle as Integer

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: Must not be 0 to have the object being valid.
(Read and Write property)

9.5.10 Constants

Constants

Constant	Value	Description
NSAttachmentCharacter	&hFFFC	This character is used to denote an attachment.

9.6 class NSTextBlockMBS

9.6.1 class NSTextBlockMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: A block of text laid out in a subregion of the text container.

Notes: A text block appears as an attribute of a paragraph, and as part of the paragraph style. The most important subclass of NSTextBlockMBS is NSTextTableBlockMBS, which represents a block of text that appears as a cell in a table. The table itself is a NSTextTable object. All NSTextBlockMBS objects reference this table, which controls their sizing and positioning.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.0](#)
- [MBS Xojo Plugins, version 22.0pr7](#)
- [MBS Xojo Plugins, version 22.0pr6](#)

Xojo Developer Magazine

- [20.2, page 10: News](#)

9.6.2 Methods

9.6.3 borderColor(edge as Integer) as NSColorMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the border color of the specified text block edge.

Notes: edge: The edge of the text block in question.

9.6.4 Constructor

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Initializes and returns an empty text block object.

9.6.5 copy as NSTextBlockMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a copy of this object.

9.6.6 `setBorderColor(color as NSColorMBS)`

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the color of all borders of the text block.

Notes: This setting has no visible effect unless the border width is larger than the default, which is 0.

See also:

- 9.6.7 `setBorderColor(color as NSColorMBS, edge as Integer)` 1020

9.6.7 `setBorderColor(color as NSColorMBS, edge as Integer)`

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the border color of the specified edge of the text block.

Notes: color: The new color.

edge: The edge whose color is to be set.

This setting has no visible effect unless the border width is larger than the default, which is 0.

See also:

- 9.6.6 `setBorderColor(color as NSColorMBS)` 1020

9.6.8 `setContentWidth(value as double, ValueType as Integer)`

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the width of the text block.

Notes: value: The new value for the width.

ValueType: The type of value being provided. This controls how val is interpreted.

9.6.9 `setValue(value as double, ValueType as Integer, dimension as Integer)`

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets a dimension of the text block.

Notes: value: The new value for the dimension.

ValueType: The type of value being provided. This controls how value is interpreted.

dimension: The dimension to set.

9.6.10 setWidth(value as double, ValueType as Integer, Layer as Integer)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the width of all edges of a specified layer of the text block.

Notes: value: The new value for the specified edge width.

type: The type of value being provided. This controls how val is interpreted.

layer: The layer of the text block to modify.

See also:

- 9.6.11 setWidth(value as double, ValueType as Integer, Layer as Integer, edge as Integer) 1021

9.6.11 setWidth(value as double, ValueType as Integer, Layer as Integer, edge as Integer)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the width of all edges of a specified layer of the text block.

Notes: value: The new value for the specified edge width.

type: The type of value being provided. This controls how val is interpreted.

layer: The layer of the text block to modify.

edge: The edge of the layer to examine.

See also:

- 9.6.10 setWidth(value as double, ValueType as Integer, Layer as Integer) 1021

9.6.12 valueForDimension(dimension as Integer) as Double

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the value of the specified text block dimension.

Notes: The value for the specified dimension. This value should be interpreted according to the value type returned by valueTypeForDimension.

9.6.13 valueTypeForDimension(dimension as Integer) as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the value type of the specified text block dimension.

Notes: The value type for the specified text block dimension. This result determines how the value for the dimension should be interpreted.

9.6.14 width(layer as Integer, edge as Integer) as double

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the width of an edge of a specified layer of the text block.

Notes: layer: The layer to examine.

edge: The edge of the layer to examine.

Returns The width of the edge of layer. This value must be interpreted according to the value type returned by `widthValueTypeForLayer()`.

9.6.15 widthValueType(layer as Integer, edge as Integer) as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the value type of an edge of a specified layer of the text block.

Notes: layer: The layer to examine.

edge: The edge of the layer to examine.

Returns the value type of the edge of layer. This determines how the value for this edge of layer should be interpreted.

9.6.16 Properties

9.6.17 backgroundColor as NSColorMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The background color of the text block.

Notes: (Read and Write property)

9.6.18 contentWidth as Double

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The width of the text block.

Notes: This property interpreted according to the value type returned by `contentWidthValueType`. (Read only property)

9.6.19 contentWidthValueType as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The type of value stored for the text block width.

Notes: This property determines how the width value should be interpreted.
(Read only property)

9.6.20 Handle as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal object reference.

Notes: (Read and Write property)

9.6.21 verticalAlignment as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The vertical alignment of the text block.

Notes: (Read and Write property)

9.6.22 Constants

Dimension

Constant	Value	Description
DimensionHeight	4	Height of the text block.
DimensionMaximumHeight	6	Maximum height of the text block.
DimensionMaximumWidth	2	Maximum width of the text block.
DimensionMinimumHeight	5	Minimum height of the text block.
DimensionMinimumWidth	1	Minimum width of the text block.
DimensionWidth	0	Width of the text block.

Layer

Constant	Value	Description
LayerBorder	0	The border of the text block.
LayerMargin	1	Margin of the text block: space surrounding the border.
LayerPadding	-1	Padding of the text block: space surrounding the content area extending to the border.

Value Types

Constant	Value	Description
ValueTypeAbsolute	0	Absolute value in points.
ValueTypePercentage	1	Percentage value (out of 100).

Vertical Alignment

Constant	Value	Description
VerticalAlignmentBaseline	3	Aligns adjacent blocks at the baseline of the first line of text in the block.
VerticalAlignmentBottom	2	Aligns adjacent blocks at their bottom.
VerticalAlignmentMiddle	1	Aligns adjacent blocks at their middle.
VerticalAlignmentTop	0	Aligns adjacent blocks at their top.

9.7 class NSTextListMBS

9.7.1 class NSTextListMBS

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: A section of text that forms a single list.

Notes: The visible elements of the list, including list markers, appear in the text as they do for lists created by hand. The list object, however, allows the list to be recognized as such by the text system. This enables automatic creation of markers and spacing. Text lists are used in HTML import and export.

Text lists appear as attributes on paragraphs, as part of the paragraph style. An `NSParagraphStyle` may have an array of text lists, representing the nested lists containing the paragraph, in order from outermost to innermost. For example, if `list1` contains four paragraphs, the middle two of which are also in the inner `list2`, then the text lists array for the first and fourth paragraphs is (`list1`), while the text lists array for the second and third paragraphs is (`list1`, `list2`).

The methods implementing this are `textLists` on `NSParagraphStyle`, and `textLists` on `NSMutableParagraphStyle`.

In addition, `NSAttributedString` has convenience methods for lists, such as `rangeOfTextList`, which determines the range covered by a list, and `itemNumberInTextList`, which determines the ordinal position within a list of a particular item.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.4](#)
- [MBS Xojo Plugins, version 23.4pr3](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.1](#)
- [MBS Xojo Plugins, version 18.1pr3](#)

Xojo Developer Magazine

- [21.6, page 8: News](#)

9.7.2 Methods

9.7.3 Constructor(format as String, OptionsMask as Integer = 0)

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Returns an initialized text list.

Notes: `format`: The marker format for the text list.

`mask`: The marker options for the text list. Values for `mask` are listed in Constants.

Returns an initialized text list.

The marker format is specified as a constant string, except for a numbering specifier, which takes the form { keyword } . The currently supported values for keyword include:

- box
- check
- circle
- diamond
- disc
- hyphen
- square
- lower-hexadecimal
- upper-hexadecimal
- octal
- lower-alpha or lower-latin
- upper-alpha or upper-latin
- lower-roman
- upper-roman
- decimal

Thus, for example, "({ decimal })" would specify the format for a list numbered (1), (2), (3), and so on, and "@ { upper-roman } " would specify the format for a list numbered I, II, III, IV, and so on. (All of these keywords are included in the Cascading Style Sheets level 3 specification.)

9.7.4 copy as NSTextListMBS

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Creates a copy of the text list.

9.7.5 markerForItemNumber(ItemNum as Integer) as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the computed value for a specific ordinal position in the list.

Notes: itemNum: The ordinal position in the list whose computed marker value is desired.

Returns the computed maker value for itemNum.

9.7.6 NSTextListMarkerBox as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.7 NSTextListMarkerCheck as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.8 NSTextListMarkerCircle as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.9 NSTextListMarkerDecimal as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.10 NSTextListMarkerDiamond as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.11 NSTextListMarkerDisc as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.12 NSTextListMarkerHyphen as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.13 NSTextListMarkerLowercaseAlpha as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.14 NSTextListMarkerLowercaseHexadecimal as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.15 NSTextListMarkerLowercaseLatin as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.16 NSTextListMarkerLowercaseRoman as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.17 NSTextListMarkerOctal as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.18 NSTextListMarkerSquare as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.19 NSTextListMarkerUppercaseAlpha as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.20 NSTextListMarkerUppercaseHexadecimal as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.21 `NSTextListMarkerUppercaseLatin` as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.22 `NSTextListMarkerUppercaseRoman` as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: One of the format texts.

Notes: Available on MacOS 10.13 or newer.

9.7.23 Properties

9.7.24 `Handle` as Integer

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: The internal reference number.

Notes: (Read and Write property)

9.7.25 `listOptions` as Integer

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the list options mask value of the receiver.

Notes: Can be `NSTextListPrependEnclosingMarker`.

(Read only property)

9.7.26 `markerFormat` as String

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Returns the marker format string used by the receiver.

Notes: (Read only property)

9.7.27 startingItemNumber as Integer

Plugin Version: 18.1, Platforms: macOS, iOS, Targets: All.

Function: Sets the starting item number for the text list.

Notes: The default value is 1. This value will be used only for ordered lists, and ignored in other cases.
(Read and Write property)

9.7.28 Constants

Constants

Constant	Value	Description
<code>NSTextListPrependEnclosingMarker</code>	1	One of the constants for options. Specifies that a nested list should include the marker for its enclosing superlist before its own marker.

9.8 class NSTextTableBlockMBS

9.8.1 class NSTextTableBlockMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: A text block that appears as a cell in a text table.

Notes: Subclass of the NSTextBlockMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.0](#)
- [MBS Xojo Plugins, version 22.0pr6](#)

9.8.2 Methods

9.8.3 Constructor

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

See also:

- 9.8.4 Constructor(table as NSTextTableMBS, startingRow as Integer, rowSpan as Integer, startingColumn as Integer, colSpan as Integer) 1032

9.8.4 Constructor(table as NSTextTableMBS, startingRow as Integer, rowSpan as Integer, startingColumn as Integer, colSpan as Integer)

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

Notes: Returns an initialized text table block.

table: The text table containing this text table block.

startingRow: The table row at which the text table block starts.

rowSpan: How many rows the text table block covers.

startingColumn: The table column at which the text table block starts.

colSpan: How many columns the text table block covers.

See also:

- 9.8.3 Constructor 1032

9.8.5 Properties

9.8.6 columnSpan as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The number of table columns spanned by this text table block.

Notes: (Read only property)

9.8.7 rowSpan as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The number of table rows spanned by this text table block.

Notes: (Read only property)

9.8.8 startingColumn as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The table column at which this text table block starts.

Notes: (Read only property)

9.8.9 startingRow as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The table row at which this text table block starts.

Notes: (Read only property)

9.8.10 table as NSTextTableMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the table containing this text table block.

Notes: (Read only property)

9.9 class NSTextTableMBS

9.9.1 class NSTextTableMBS

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: An object that represents a text table as a whole.

Notes: A text table is responsible for laying out and drawing the text table blocks it contains, and it maintains the basic parameters of the table.

Subclass of the NSTextBlockMBS class.

Blog Entries

- [News from the MBS Xojo Plugins Version 22.0](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 22.0](#)
- [MBS Xojo Plugins, version 22.0pr6](#)

9.9.2 Methods

9.9.3 Constructor

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The constructor.

9.9.4 Properties

9.9.5 collapsesBorders as Boolean

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: A Boolean value indicating whether the text table borders are collapsible.

Notes: The value of this property is true when the text table borders are collapsible.
(Read and Write property)

9.9.6 hidesEmptyCells as Boolean

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: A Boolean value indicating whether the text table hides empty cells.

Notes: The value of this property is true when the text table hides empty cells. If empty cells are hidden, locations with empty cells allow the background of the enclosing block or text container to show through.

(Read and Write property)

9.9.7 layoutAlgorithm as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The text table layout algorithm.

Notes: (Read and Write property)

9.9.8 numberOfColumns as Integer

Plugin Version: 22.0, Platform: macOS, Targets: Desktop, Console & Web.

Function: The number of columns in the text table.

Notes: (Read and Write property)

9.9.9 Constants

Text table layout algorithm

Constant	Value	Description
LayoutAlgorithmAutomatic	0	Specifies automatic layout algorithm
LayoutAlgorithmFixed	1	Specifies fixed layout algorithm

9.10 class NSUndoManagerMBS

9.10.1 class NSUndoManagerMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: NSUndoManager is a general-purpose recorder of operations for undo and redo.

Notes: You register an undo operation by specifying the object that's changing (or the owner of that object), along with a method to invoke to revert its state, and the arguments for that method. When performing undo an NSUndoManager saves the operations reverted so that you can redo the undos. If used in a Cocoa Application Kit-based application, NSUndoManager groups all operations within a single cycle of the run loop, so that performing an undo reverts all changes that occurred during the cycle.

NSUndoManager is implemented as a class of the Foundation framework because executables other than applications might want to revert changes to their states. For example, you might have an interactive command-line tool with undo and redo commands, or there could be distributed object implementations that can revert operations "over the wire." However, users typically see undo and redo as application features. The Application Kit implements undo and redo in its NSTextView object and makes it easy to implement it in objects along the responder chain.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

9.10.2 Methods

9.10.3 beginUndoGrouping

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Marks the beginning of an undo group.

Notes: All individual undo operations before a subsequent endUndoGrouping message are grouped together and reversed by a later undo message. By default undo groups are begun automatically at the start of the event loop, but you can begin your own undo groups with this method, and nest them within other groups.

9.10.4 canRedo as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver has any actions to redo.

9.10.5 canUndo as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver has any actions to undo.

Notes: The return value does not mean you can safely invoke `undo` or `undoNestedGroup`—you may have to close open undo groups first.

9.10.6 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

9.10.7 `disableUndoRegistration`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Disables the recording of undo operations.

Notes: This method can be invoked multiple times by multiple clients. The `enableUndoRegistration` method must be invoked an equal number of times to re-enable undo registration.

9.10.8 `enableUndoRegistration`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Enables the recording of undo operations.

Notes: Because undo registration is enabled by default, it is often used to balance a prior `disableUndoRegistration` message. Undo registration isn't actually re-enabled until an `enable` message balances the last `disable` message in effect. Raises an `NSInternalInconsistencyException` if invoked while no `disableUndoRegistration` message is in effect.

9.10.9 `endUndoGrouping`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Marks the end of an undo group.

Notes: All individual undo operations back to the matching `beginUndoGrouping` message are grouped together and reversed by a later `undo` or `undoNestedGroup` message. Undo groups can be nested, thus providing functionality similar to nested transactions. Raises an `NSInternalInconsistencyException` if there's no `beginUndoGrouping` message in effect.

9.10.10 groupingLevel as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the number of nested undo groups (or redo groups, if Redo was invoked last) in the current event loop.

Notes: An integer indicating the number of nested groups. If 0 is returned, there is no open undo or redo group.

9.10.11 isRedoing as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver is in the process of performing its redo method.

Notes: True if the method is being performed, otherwise false.

9.10.12 isUndoing as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver is in the process of performing its undo or undoNestedGroup method.

Notes: True if the method is being performed, otherwise false.

9.10.13 isUndoRegistrationEnabled as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the recording of undo operations is enabled.

Notes: True if registration is enabled; otherwise, false.

Undo registration is enabled by default.

9.10.14 redo

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Performs the operations in the last group on the redo stack, if there are any, recording them on the undo stack as a single group.

Notes: Raises an `NSInternalInconsistencyException` if the method is invoked during an undo operation.

9.10.15 redoActionName as string

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the name identifying the redo action.

Notes: The redo action name. Returns an empty string if no action name has been assigned or if there is nothing to redo.

For example, if the menu title is "Redo Delete," the string returned is "Delete."

9.10.16 redoMenuItemTitle as string

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the complete title of the Redo menu command, for example, "Redo Paste."

Notes: Returns "Redo" if no action name has been assigned or an empty string if there is nothing to redo.

9.10.17 redoMenuTitleForUndoActionName(actionName as string) as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the complete, localized title of the Redo menu command for the action identified by the given name.

Notes: Override this method if you want to customize the localization behavior. This method is invoked by redoMenuItemTitle.

Available in Mac OS X v10.0 and later.

9.10.18 removeAllActions

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Clears the undo and redo stacks and re-enables the receiver.

9.10.19 setActionName(actionName as string)

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Sets the name of the action associated with the Undo or Redo command.

Notes: If actionName is an empty string, the action name currently associated with the menu command is

removed. There is no effect if `actionName` is `""`.

9.10.20 `undo`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Closes the top-level undo group if necessary and invokes `undoNestedGroup`.

Notes: This method also invokes `endUndoGrouping` if the nesting level is 1. Raises an `NSInternalInconsistencyException` if more than one undo group is open (that is, if the last group isn't at the top level).

9.10.21 `undoActionName` as string

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the name identifying the undo action.

Notes: The undo action name. Returns an empty string if no action name has been assigned or if there is nothing to undo.

For example, if the menu title is "Undo Delete," the string returned is "Delete."

9.10.22 `undoMenuItemTitle` as string

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the complete title of the Undo menu command, for example, "Undo Paste."

Notes: Returns "Undo" if no action name has been assigned or `""` if there is nothing to undo.

9.10.23 `undoMenuItemTitleForUndoActionName(actionName as string)` as string

Plugin Version: 9.6, Platform: macOS, Targets: Desktop only.

Function: Returns the complete, localized title of the Undo menu command for the action identified by the given name.

Notes: Override this method if you want to customize the localization behavior. This method is invoked by `undoMenuItemTitle`.

9.10.24 `undoNestedGroup`

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Performs the undo operations in the last undo group (whether top-level or nested), recording the operations on the redo stack as a single group.

Notes: Raises an `NSInternalInconsistencyException` if any undo operations have been registered since the last `enableUndoRegistration` message.

9.10.25 Properties

9.10.26 Handle as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: The internal uses reference to the `NSUndoManager` object.

Notes: (Read and Write property)

9.10.27 groupsByEvent as boolean

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value that indicates whether the receiver automatically creates undo groups around each pass of the run loop.

Notes: True if the receiver automatically creates undo groups around each pass of the run loop, otherwise false.

(Read and Write computed property)

9.10.28 levelsOfUndo as Integer

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns the maximum number of top-level undo groups the receiver holds.

Notes: An integer specifying the number of undo groups. A limit of 0 indicates no limit, so old undo groups are never dropped.

When ending an undo group results in the number of groups exceeding this limit, the oldest groups are dropped from the stack. The default is 0.

(Read and Write computed property)

Chapter 10

Cocoa Toolbar

Chapter 11

Controls

11.1 class Control

11.1.1 class Control

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

Function: The built in Control class in Xojo.

11.1.2 Methods

11.1.3 NSViewMBS as NSViewMBS

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

Function: Creates a NSViewMBS object for the given control.

Example:

```
MsgBox PushButton1.NSViewMBS.className
```

Notes: This way you can manipulate Cocoa controls directly.

For a pushbutton you may want to cast the NSViewMBS to NSButtonMBS to have more options.

11.2 class DesktopControl

11.2.1 class DesktopControl

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

Function: The built in Control class in Xojo.

11.2.2 Methods

11.2.3 NSViewMBS as NSViewMBS

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

Function: Creates a NSViewMBS object for the given control.

Example:

```
MsgBox PushButton1.NSViewMBS.className
```

Notes: This way you can manipulate Cocoa controls directly.

For a pushbutton you may want to cast the NSViewMBS to NSButtonMBS to have more options.

Chapter 12

Currency, Date and Time Format

12.1 class NSLocaleDateMBS

12.1.1 class NSLocaleDateMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: A class for a date format.

Notes: Should be used with the NSLocaleMBS class, but can be used on its own, too.

Blog Entries

- [MBS Xojo Plugins, version 22.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr4](#)

12.1.2 Methods

12.1.3 Constructor

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The default constructor.

Notes: Creates a different object than the once you get by the NSLocaleMBS class. But this date information you get than works on Mac OS X 10.0. The NSLocaleMBS class is Mac OS X 10.4 only.

See also:

- 12.1.4 Constructor(locale as NSLocaleMBS)

1048

12.1.4 Constructor(locale as NSLocaleMBS)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The constructor.

Example:

```
dim n1 as new NSLocaleDateMBS
dim n2 as new NSLocaleDateMBS(NSLocaleMBS.currentLocale)
dim n3 as new NSLocaleDateMBS(NSLocaleMBS.systemLocale)
```

break // see differences in debugger

Notes: Initializes the object for a given locale.

See also:

- 12.1.3 Constructor

1047

12.1.5 eraSymbols as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the array of era symbols.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.eraSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "v. Chr., n. Chr." (in Germany)
```

```
// shows for example "BC, AD" (in USA)
```

```
// shows for example "av. J.-C., ap. J.-C." (in France)
```

Notes: for example, "BCE", "CE" in the USA.

On error returns an empty array.

12.1.6 longEraSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the long era symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.longEraSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "v. Chr., n. Chr." (in Germany)
// shows for example "Before Christ, Anno Domini" (in USA)
// shows for example "avant J/©sus-Christ, apr/©s J/©sus-Christ" (in France)
```

Notes: An array containing strings representing the era symbols for the receiver (for example, { "Before Common Era", "Common Era" }).

Available in Mac OS X v10.5 and later.

12.1.7 monthSymbols as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the month symbols.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.monthSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "Januar, Februar, M/©rz, April, Mai, Juni, Juli, August, September, Oktober, November, Dezember" (in Germany)
// shows for example "January, February, March, April, May, June, July, August, September, October, November, December" (in USA)
// shows for example "janvier, f/©vrier, mars, avril, mai, juin, juillet, ao/©t, septembre, octobre, novembre, d/©cembre" (in France)
```

Notes: On error returns an empty array.

12.1.8 quarterSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the quarter symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.quarterSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "1. Quartal, 2. Quartal, 3. Quartal, 4. Quartal" (in Germany)
// shows for example "1st quarter, 2nd quarter, 3rd quarter, 4th quarter" (in USA)
// shows for example "1er trimestre, 2e trimestre, 3e trimestre, 4e trimestre" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.9 shortMonthSymbols as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the array of short month.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortMonthSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "Jan, Feb, Mrz, Apr, Mai, Jun, Jul, Aug, Sep, Okt, Nov, Dez" (in Germany)
// shows for example "Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec" (in USA)
// shows for example "janv., f√©vr., mars, avr., mai, juin, juil., ao√ºt, sept., oct., nov., d√©c." (in France)
```

Notes: On error returns an empty array.

12.1.10 shortQuarterSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the short quarter symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortQuarterSymbols
```

```

MsgBox join(s, ", ")

// shows for example "Q1, Q2, Q3, Q4" (in Germany)
// shows for example "Q1, Q2, Q3, Q4" (in USA)
// shows for example "T1, T2, T3, T4" (in France)

```

Notes: Available in Mac OS X v10.5 and later.

12.1.11 shortStandaloneMonthSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the short standalone month symbols for the receiver.

Example:

```

dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortStandaloneMonthSymbols

```

```

MsgBox join(s, ", ")

```

```

// shows for example "Jan, Feb, M√§r, Apr, Mai, Jun, Jul, Aug, Sep, Okt, Nov, Dez" (in Germany)
// shows for example "Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec" (in USA)
// shows for example "janv., f√©vr., mars, avr., mai, juin, juil., ao√ºt, sept., oct., nov., d√©c." (in France)

```

Notes: Available in Mac OS X v10.5 and later.

12.1.12 shortStandaloneQuarterSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the short standalone quarter symbols for the receiver.

Example:

```

dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortStandaloneQuarterSymbols

```

```

MsgBox join(s, ", ")

```

```

// shows for example "Q1, Q2, Q3, Q4" (in Germany)
// shows for example "Q1, Q2, Q3, Q4" (in USA)

```

```
// shows for example "T1, T2, T3, T4" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.13 shortStandaloneWeekdaySymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the array of short standalone weekday symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortStandaloneWeekdaySymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "So., Mo., Di., Mi., Do., Fr., Sa." (in Germany)
// shows for example "Sun, Mon, Tue, Wed, Thu, Fri, Sat" (in USA)
// shows for example "dim., lun., mar., mer., jeu., ven., sam." (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.14 shortWeekdaySymbols as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Sets the short weekday symbols.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.shortWeekdaySymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "So., Mo., Di., Mi., Do., Fr., Sa." (in Germany)
// shows for example "Sun, Mon, Tue, Wed, Thu, Fri, Sat" (in USA)
// shows for example "dim., lun., mar., mer., jeu., ven., sam." (in France)
```

Notes: On error returns an empty array.

12.1.15 standaloneMonthSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the standalone month symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.standaloneMonthSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "Januar, Februar, März, April, Mai, Juni, Juli, August, September, Oktober, November, Dezember" (in Germany)
```

```
// shows for example "January, February, March, April, May, June, July, August, September, October, November, December" (in USA)
```

```
// shows for example "janvier, février, mars, avril, mai, juin, juillet, août, septembre, octobre, novembre, décembre" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.16 standaloneQuarterSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the standalone quarter symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.standaloneQuarterSymbols
```

```
MsgBox join(s, ", ")
```

```
// shows for example "1. Quartal, 2. Quartal, 3. Quartal, 4. Quartal" (in Germany)
```

```
// shows for example "1st quarter, 2nd quarter, 3rd quarter, 4th quarter" (in USA)
```

```
// shows for example "1er trimestre, 2e trimestre, 3e trimestre, 4e trimestre" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.17 standaloneWeekdaySymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the array of standalone weekday symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.standaloneWeekdaySymbols

MsgBox join(s,", ")

// shows for example "Sonntag, Montag, Dienstag, Mittwoch, Donnerstag, Freitag, Samstag" (in Germany)
// shows for example "Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday" (in USA)
// shows for example "dimanche, lundi, mardi, mercredi, jeudi, vendredi, samedi" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.18 veryShortMonthSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the very short month symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.veryShortMonthSymbols

MsgBox join(s,", ")

// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in Germany)
// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in USA)
// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.19 veryShortStandaloneMonthSymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the very short month symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.veryShortStandaloneMonthSymbols

MsgBox join(s,", ")

// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in Germany)
// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in USA)
// shows for example "J, F, M, A, M, J, J, A, S, O, N, D" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.20 veryShortStandaloneWeekdaySymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the array of very short standalone weekday symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.veryShortStandaloneWeekdaySymbols

MsgBox join(s,", ")

// shows for example "S, M, D, M, D, F, S" (in Germany)
// shows for example "S, M, T, W, T, F, S" (in USA)
// shows for example "D, L, M, M, J, V, S" (in France)
```

Notes: Available in Mac OS X v10.5 and later.

12.1.21 veryShortWeekdaySymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the array of very short weekday symbols for the receiver.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
```

```

dim s(-1) as string = n.veryShortWeekdaySymbols

MsgBox join(s, ", ")

// shows for example "S, M, D, M, D, F, S" (in Germany)
// shows for example "S, M, T, W, T, F, S" (in USA)
// shows for example "D, L, M, M, J, V, S" (in France)

```

Notes: Available in Mac OS X v10.5 and later.

12.1.22 weekdaySymbols as string()

Plugin Version: 10.0, Platform: macOS, Targets: All.

Function: Returns the array of weekday symbols for the receiver.

Example:

```

dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong
dim s(-1) as string = n.weekdaySymbols

```

```

MsgBox join(s, ", ")

// shows for example "Sonntag, Montag, Dienstag, Mittwoch, Donnerstag, Freitag, Samstag" (in Germany)
// shows for example "Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday" (in USA)
// shows for example "dimanche, lundi, mardi, mercredi, jeudi, vendredi, samedi" (in France)

```

Notes: Available in Mac OS X v10.4 and later.

12.1.23 Properties

12.1.24 AMSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the AM symbol.

Example:

```

dim l as new NSLocaleMBS
dim n as NSLocaleDateMBS = l.DateLong

```

```

MsgBox n.AMSymbol // shows for example "vorm." (in Germany)

```

Notes: (Read only property)

12.1.25 dateFormat as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the date format string,

Example:

```
dim l as new NSLocaleMBS
```

```
dim n as NSLocaleDateMBS = l.DateLong
```

```
MsgBox n.dateFormat // shows for example: "d. MMMM yyyy" (in Germany)
```

Notes: Format of this string (if object was made with new):

http://developer.apple.com/documentation/Cocoa/Conceptual/DataFormatting/Articles/dfDateFormatter-Syntax.html#//apple_ref/doc/uid/20000194

If you got this object from NSLocale, it uses a different format.

(Read only property)

12.1.26 PMSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the PM symbol.

Example:

```
dim l as new NSLocaleMBS
```

```
dim n as NSLocaleDateMBS = l.DateLong
```

```
MsgBox n.PMSymbol // shows for example "nachm." (in Germany)
```

Notes: (Read only property)

12.2 class NSLocaleMBS

12.2.1 class NSLocaleMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The class for CoCoas localization data.

Example:

```
dim n as NSLocaleMBS
```

```
n=NSLocaleMBS.currentLocale
```

```
MsgBox n.CountryCode
```

Notes: Available in Mac OS X v10.4 and later.

Blog Entries

- [MBS Xojo Plugins, version 22.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr4](#)
- [MBS Real Studio Plugins, version 12.5pr8](#)
- [MBS REALbasic plug-in 9.6](#)

12.2.2 Methods

12.2.3 autoupdatingCurrentLocale as NSLocaleMBS

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns the current logical locale for the current user.

Notes: The current logical locale for the current user. The locale is formed from the settings for the current user's chosen system locale overlaid with any custom settings the user has specified in System Preferences.

The object always reflects the current state of the current user's locale settings.

Settings you get from this locale do change as the user's settings change (contrast with `currentLocale`).

Note that if you cache values based on the locale or related information, those caches will of course not be automatically updated by the updating of the locale object.

Available in Mac OS X v10.5 and later.

12.2.4 availableLocaleIdentifiers as string()

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns an array of strings, each of which identifies a locale available on the system.

Example:

```
MsgBox join(NSLocaleMBS.availableLocaleIdentifiers,", ")
```

```
// shows "zh_Hans_CN, sq_AL, he_IL, uz_Latn, en_CA, zh, kk_Cyrl, sr_Cyrl,
// fr_CA, sr_Cyrl_RS, fr_LU, bg_BG, es_ES, da_DK, el_CY, ja_JP, kok_IN,
// mt_MT, ar_JO, de_LU, uz_Cyrl, pt_PT, af, ms_MY, nl_BE, es_US, en_PH,
// es_HN, be, sr_Latn_BA, es_CR, ca, vi_VN, zh_Hant_HK, bg, gu_IN, si_LK,
// am, or_IN, en_ZA, ar_YE, da, uk_UA, ar_OM, es_PE, cs_CZ, kk_Cyrl_KZ,
// de_AT, lv_LV, en_NZ, ar_BH, ar, en_AU, az_Cyrl, de, as, bn, zh_Hans_SG,
// es_PY, es_EC, ne_IN, pa_Guru_IN, fa, sw_KE, es_AR, az, fr_FR,
// en_US_POSIX, sr_Cyrl_BA, cs, zh_Hans, eu_ES, ga, en_PK, ar_LB, el,
// zh_Hant_TW, et_EE, fi, en, uz_Arab_AF, sw_TZ, it_CH, en_ZW, ha, eo,
// ar_SA, cy, th_TH, bn_BD, te_IN, ml_IN, he, es, fo, en_IN, et, om_KE,
// haw, eu, gl, ta_IN, zh_Hans_MO, id, fr, hi, ja, es_SV, hu_HU, en_SG,
// nn_NO, az_Latn, ii, id_ID, es_DO, fr_CH, ka, es_UY, en_MT, ar_QA, ar_TN,
// gu, de_CH, gv, sk_SK, hr, kn_IN, de_LI, ar_EG, el_GR, ar_SD, hu, af_NA,
// zh_Hant, ti_ER, ar_LY, is, ka_GE, it, om_ET, hy, ii_CN, hr_HR, ps_AF,
// pa_Arab_PK, es_NI, en_TT, pa_Guru, kk, kl, be_BY, km, zh_Hans_HK, kn,
// sv_SE, es_BO, ko, nb, ne_NP, pa_Arab, fr_MC, is_IS, ne, ar_SY, mk,
// uz_Arab, so_KE, ml, it_IT, hy_AM, sl_SI, ti_ET, uz_Latn_UZ, ar_AE,
// lt_LT, kw, pa, lt, nl, lv, so_SO, mr, nn, ru_RU, ms, ga_IE, fo_FO, mt,
// om, ar_MA, ms_BN, pl, ha_Latn, kok, mk_MK, or, uz_Cyrl_UZ, so_ET, de_DE,
// gl_ES, en_BW, en_JM, ar_IQ, ps, az_Latn_AZ, ta, pt, so_DJ, nl_NL, af_ZA,
// nb_NO, en_HK, fr_SN, si, km_KH, te, ro, ko_KR, mr_IN, sk, en_BE, en_MH,
// sl, th, ti, fr_BE, es_CL, pt_BR, en_VI, es_VE, sr_Latn_ME, so, hi_IN,
// as_IN, ru, de_BE, sq, en_GB, sr, fa_AF, es_MX, kw_GB, es_PR, ro_RO, uk,
// cy_GB, en_NA, fi_FI, ca_ES, az_Cyrl_AZ, en_IE, gv_GB, sv, tr, sw, vi,
// en_BZ, en_US, hy_AM_REVISIED, ru_UA, ur_PK, ur, sr_Latn_RS, am_ET, bn_IN,
// ar_KW, haw_US, sr_Cyrl_ME, es_PA, es_CO, kl_GL, sr_Latn, uz, ur_IN,
// zh_Hant_MO, ha_Latn_NG, es_GT, tr_TR, fa_IR, pl_PL, sv_FI, ar_DZ"
```

Notes: Available in Mac OS X v10.4 and later.

12.2.5 canonicalLanguageIdentifierFromString(s as string) as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: eturns a canonical language identifier by mapping an arbitrary locale identification string to the canonical identifier.

Notes: s: A string representation of an arbitrary locale identifier.

Returns a string that represents the canonical language identifier for the specified arbitrary locale identifier. Available in Mac OS X v10.6 and later.

12.2.6 canonicalLocaleIdentifierFromString(s as string) as string

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the canonical identifier for a given locale identification string.

Notes: Available in Mac OS X v10.4 and later.

12.2.7 characterDirectionForLanguage(isoLangCode as string) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the character direction for the specified ISO language code.

Notes: Returns the character direction for the language. See constants for possible values. If the appropriate direction can't be determined NSLocaleLanguageDirectionUnknown is returned.

Available in Mac OS X v10.6 and later.

12.2.8 commonISOCurrencyCodes as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns an array of common ISO currency codes.

Example:

```
MsgBox join(NSLocaleMBS.commonISOCurrencyCodes, ", ")
```

```
// shows "AED, AFN, ALL, AMD, ANG, AOA, ARS, AUD, AWG, AZN,
// BAM, BBD, BDT, BGN, BHD, BIF, BMD, BND, BOB, BRL, BSD, BTN,
// BWP, BYR, BZD, CAD, CDF, CHF, CLP, CNY, COP, CRC, CUP, CVE,
// CZK, DJF, DKK, DOP, DZD, EEK, EGP, ERN, ETB, EUR, FJD, FKP,
// GBP, GEL, GHS, GIP, GMD, GNF, GTQ, GWP, GYD, HKD, HNL, HRK,
// HTG, HUF, IDR, ILS, INR, IQD, IRR, ISK, JMD, JOD, JPY, KES,
// KGS, KHR, KMF, KPW, KRW, KWD, KYD, KZT, LAK, LBP, LKR, LRD,
// LSL, LTL, LVL, LYD, MAD, MDL, MGA, MKD, MMK, MNT, MOP, MRO,
// MUR, MVR, MWK, MXN, MYR, MZE, MZN, NAD, NGN, NIO, NOK, NPR,
// NZD, OMR, PAB, PEN, PGK, PHP, PKR, PLN, PYG, QAR, RON, RSD,
// RUB, RWF, SAR, SBD, SCR, SDG, SEK, SGD, SHP, SKK, SLL, SOS,
// SRD, STD, SVC, SYP, SZL, THB, TJS, TMM, TND, TOP, TRY, TTD,
// TWD, TZS, UAH, UGX, USD, UYU, UZS, VEF, VND, VUV, WST, XAF,
```

```
// XCD, XOF, XPF, YER, ZAR, ZMK, ZWD”
```

Notes: Common codes may include, for example, AED, AUD, BZD, DKK, EUR, GBP, JPY, KES, MXN, OMR, STD, USD, XCD, and ZWD.

Available in Mac OS X v10.5 and later.

12.2.9 Constructor

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The default constructor.

Notes: Available in Mac OS X v10.4 and later.

Loads the same values for this object as if you just take the object from NSLocaleMBS.currentLocale.

See also:

- 12.2.10 Constructor(Identifier as string)

1061

12.2.10 Constructor(Identifier as string)

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: The constructor.

Example:

```
dim a(-1) as string = NSLocalembs.availableLocaleIdentifiers
```

```
// show all identifiers:
```

```
dim s(-1) as string
```

```
s.Append "All available Locale Identifiers:"
```

```
dim c as Integer = UBound(a)
```

```
for i as Integer=0 to c
```

```
dim identifier as string=a(i)
```

```
s.Append identifier
```

```
next
```

```
MsgBox Join(s)
```

```
// now show the currency symbols:
```

```
redim s(-1)
```

```
s.Append "All Currency symbols:"
```

```

c = UBound(a)
for i as Integer=0 to c
dim identifier as string=a(i)
dim n as new NSLocaleMBS(identifier)
s.Append n.CurrencySymbol
next

MsgBox Join(s)

```

Notes: Available in Mac OS X v10.4 and later.

Pass in the country identifier which you get with the availableLocaleIdentifiers function.
See also:

- 12.2.9 Constructor

1061

12.2.11 currentLocale as NSLocaleMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the logical locale for the current user.

Notes: Available in Mac OS X v10.4 and later.

The logical locale for the current user. The locale is formed from the settings for the current user's chosen system locale overlaid with any custom settings the user has specified in System Preferences.

This method may return a retained cached object.

Discussion:

Settings you get from this locale do not change as System Preferences are changed so that your operations are consistent. Typically you perform some operations on the returned object and then allow it to be disposed of. Moreover, since the returned object may be cached, you do not need to hold on to it indefinitely.

12.2.12 displayName(key as string, value as string) as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns the display name for the given value.

Example:

```

dim displayNameString as string

// The first uses the fr_FR locale.

```

```

dim frLocale as new NSLocaleMBS("fr_FR")

displayNameString = frLocale.displayName(frLocale.NSLocaleIdentifier, "fr_FR")
MsgBox "display name for fr_FR in fr_FR: "+displayNameString
// shows: "display name for fr_FR in fr_FR: français (France)"

displayNameString = frLocale.displayName(frLocale.NSLocaleIdentifier, "en_US")
MsgBox "display name for en_US in fr_FR: "+displayNameString
// shows: "display name for en_US in fr_FR: anglais (États-Unis)"

// The following example uses the en_GB locale.

dim gbLocale as new NSLocaleMBS("en_US")

displayNameString = gbLocale.displayName(frLocale.NSLocaleIdentifier, "fr_FR")
MsgBox "display name for fr_FR in en_US: "+displayNameString
// shows: "display name for fr_FR in en_US: French (France)"

displayNameString = gbLocale.displayName(frLocale.NSLocaleIdentifier, "en_US")
MsgBox "display name for en_US in en_US: "+displayNameString
// shows: "display name for en_US in en_US: English (United States)"

```

Notes: Not all locale property keys have values with display name values.

You can use the NSLocaleIdentifier key to get the name of a locale in the language of another locale, as illustrated in the example code above.

12.2.13 ExemplarCharacterSet as Variant

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: The exemplar character set for the locale.

Example:

```

dim n as NSLocaleMBS = NSLocaleMBS.currentLocale

MsgBox n.ExemplarCharacterSet.StringValue
// shows "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzÿŃñúüŸ√°" in Ger-
many

```

Notes: Returns a NSCharacterSetMBS object.
Available in Mac OS X v10.4 and later.

12.2.14 GetString(key as string) as string

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns a string for one of the NSLocale keys.

Notes: Available in Mac OS X v10.4 and later.

This keys can be found in the NSLocale reference from Apple.

12.2.15 ISOCountryCodes as string()

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns an array of strings that represents all known legal country codes.

Example:

```
MsgBox join(NSLocaleMBS.ISOCountryCodes,", ")
```

```
// shows "AD, AE, AF, AG, AI, AL, AM, AN, AO, AQ, AR, AS, AT, AU,
// AW, AX, AZ, BA, BB, BD, BE, BF, BG, BH, BI, BJ, BL, BM, BN, BO,
// BR, BS, BT, BV, BW, BY, BZ, CA, CC, CD, CF, CG, CH, CI, CK, CL,
// CM, CN, CO, CR, CU, CV, CX, CY, CZ, DE, DJ, DK, DM, DO, DZ, EC,
// EE, EG, EH, ER, ES, ET, FI, FJ, FK, FM, FO, FR, GA, GB, GD, GE,
// GF, GG, GH, GI, GL, GM, GN, GP, GQ, GR, GS, GT, GU, GW, GY, HK,
// HM, HN, HR, HT, HU, ID, IE, IL, IM, IN, IO, IQ, IR, IS, IT, JE,
// JM, JO, JP, KE, KG, KH, KI, KM, KN, KP, KR, KW, KY, KZ, LA, LB,
// LC, LI, LK, LR, LS, LT, LU, LV, LY, MA, MC, MD, ME, MF, MG, MH,
// MK, ML, MM, MN, MO, MP, MQ, MR, MS, MT, MU, MV, MW, MX, MY, MZ,
// NA, NC, NE, NF, NG, NI, NL, false, NP, NR, NU, NZ, OM, PA, PE, PF,
// PG, PH, PK, PL, PM, PN, PR, PS, PT, PW, PY, QA, RE, RO, RS, RU,
// RW, SA, SB, SC, SD, SE, SG, SH, SI, SJ, SK, SL, SM, SN, SO, SR,
// ST, SV, SY, SZ, TC, TD, TF, TG, TH, TJ, TK, TL, TM, TN, TO, TR,
// TT, TV, TW, TZ, UA, UG, UM, US, UY, UZ, VA, VC, VE, VG, VI, VN,
// VU, WF, WS, YE, YT, ZA, ZM, ZW"
```

Notes: Available in Mac OS X v10.4 and later.

Note that many of country codes do not have any supporting locale data in Mac OS X.

12.2.16 ISOCurrencyCodes as string()

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns an array of strings that represents all known legal ISO currency codes.

Example:

```
MsgBox join(NSLocaleMBS.ISOCurrencyCodes," ")

// shows "ADP, AED, AFA, AFN, ALK, ALL, AMD, ANG, AOA, AOK, AON, AOR,
// ARA, ARP, ARS, ATS, AUD, AWG, AZM, AZN, BAD, BAM, BBD, BDT, BEC,
// BEF, BEL, BGL, BGM, BGN, BHD, BIF, BMD, BND, BOB, BOP, BOV, BRB,
// BRC, BRE, BRL, BRN, BRR, BSD, BTN, BUK, BWP, BYB, BYR, BZD, CAD,
// CDF, CHE, CHF, CHW, CLF, CLP, CNX, CNY, COP, COU, CRC, CSD, CSK,
// CUP, CVE, CYP, CZK, DDM, DEM, DJF, DKK, DOP, DZD, ECS, ECV, EEK,
// EGP, EQE, ERN, ESA, ESB, ESP, ETB, EUR, FIM, FJD, FKP, FRF, GBP,
// GEK, GEL, GHC, GHS, GIP, GMD, GNF, GNS, GQE, GRD, GTQ, GWE, GWP,
// GYD, HKD, HNL, HRD, HRK, HTG, HUF, IDR, IEP, ILP, ILS, INR, IQD,
// IRR, ISK, ITL, JMD, JOD, JPY, KES, KGS, KHR, KMF, KPW, KRW, KWD,
// KYD, KZT, LAK, LBP, LKR, LRD, LSL, LSM, LTL, LTT, LUC, LUF, LUL,
// LVL, LVR, LYD, MAD, MAF, MDL, MGA, MGF, MKD, MLF, MMK, MNT, MOP,
// MRO, MTL, MTP, MUR, MVR, MWK, MXN, MXP, MXV, MYR, MZE, MZM, MZN,
// NAD, NGN, NIC, NIO, NLG, NOK, NPR, NZD, OMR, PAB, PEI, PEN, PES,
// PGK, PHP, PKR, PLN, PLZ, PTE, PYG, QAR, RHD, ROL, RON, RSD, RUB,
// RUR, RWF, SAR, SBD, SCR, SDD, SDG, SDP, SEK, SGD, SHP, SIT, SKK,
// SLL, SOS, SRD, SRG, STD, SUR, SVC, SYP, SZL, THB, TJR, TJS, TMM,
// TND, TOP, TPE, TRL, TRY, TTD, TWD, TZS, UAH, UAK, UGS, UGX, USD,
// USN, USS, UYP, UYI, UYU, UZS, VEB, VEF, VND, VUV, WST, XAF, XAG,
// XAU, XBA, XBB, XBC, XBD, XCD, XDR, XEU, XFO, XFU, XOF, XPD, XPF,
// XPT, XRE, XTS, XXX, YDD, YER, YUD, YUM, YUN, ZAL, ZAR, ZMK, ZRN,
// ZRZ, ZWD"
```

Notes: Available in Mac OS X v10.4 and later.

Note that some of the currency codes may not have any supporting locale data in Mac OS X.

12.2.17 ISOLanguageCodes as string()

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns an array of strings that represents all known legal ISO language codes.

Example:

```
MsgBox join(NSLocaleMBS.ISOLanguageCodes," ")

// shows "aa, ab, ace, ach, ada, ady, ae, af, afa, afh, ain, ak, akk,
// ale, alg, alt, am, an, ang, anp, apa, ar, arc, arn, arp, art, arw,
// as, ast, ath, aus, av, awa, ay, az, ba, bad, bai, bal, ban, bas,
// bat, be, bej, bem, ber, bg, bh, bho, bi, bik, bin, bla, bm, bn, bnt,
```

```
// bo, br, bra, bs, btk, bua, bug, byn, ca, cad, cai, car, cau, cch,
// ce, ceb, cel, ch, chb, chg, chk, chm, chn, cho, chp, chr, chy, cmc,
// co, cop, cpe, cpf, cpp, cr, crh, crp, cs, csb, cu, cus, cv, cy, da,
// dak, dar, day, de, del, den, dgr, din, doi, dra, dsb, dua, dum, dv,
// dyu, dz, ee, efi, egy, eka, el, elx, en, enm, eo, es, et, eu, ewo,
// fa, fan, fat, ff, fi, fil, fiu, fj, fo, fon, fr, frm, fro, frr, frs,
// fur, fy, ga, gaa, gay, gba, gd, gem, gez, gil, gl, gmh, gn, goh,
// gon, gor, got, grb, grc, gsw, gu, gv, gwi, ha, hai, haw, he, hi,
// hil, him, hit, hmn, ho, hr, hsb, ht, hu, hup, hy, hz, ia, iba, id,
// ie, ig, ii, ijo, ik, ilo, inc, ine, inh, io, ira, iro, is, it, iu,
// ja, jbo, jpr, jrb, jv, ka, kaa, kab, kac, kaj, kam, kar, kaw, kbd,
// kcg, kfo, kg, kha, khi, kho, ki, kj, kk, kl, km, kmb, kn, ko, kok,
// kos, kpe, kr, krc, krl, kro, kru, ks, ku, kum, kut, kv, kw, ky, la,
// lad, lah, lam, lb, lez, lg, li, ln, lo, lol, loz, lt, lu, lua, lui,
// lun, luo, lus, lv, mad, mag, mai, mak, man, map, mas, mdf, mdr, men,
// mg, mga, mh, mi, mic, min, mis, mk, mkh, ml, mn, mnc, mni, mno, mo,
// moh, mos, mr, ms, mt, mul, mun, mus, mwl, mwr, my, myn, myv, na,
// nah, nai, nap, nb, nd, nds, ne, new, ng, nia, nic, niu, nl, nn, no,
// nog, non, nqo, nr, nso, nub, nv, nwc, ny, nym, nyn, nyo, nzi, oc,
// oj, om, or, os, osa, ota, oto, pa, paa, pag, pal, pam, pap, pau,
// peo, phi, phn, pi, pl, pon, pra, pro, ps, pt, qu, raj, rap, rar, rm,
// rn, ro, roa, rom, ru, rup, rw, sa, sad, sah, sai, sal, sam, sas,
// sat, sc, scn, sco, sd, se, sel, sem, sg, sga, sgn, shn, si, sid,
// sio, sit, sk, sl, sla, sm, sma, smi, smj, smn, sms, sn, snk, so,
// sog, son, sq, sr, srn, srr, ss, ssa, st, su, suk, sus, sux, sv, sw,
// syc, syr, ta, tai, te, tem, ter, tet, tg, th, ti, tig, tiv, tk, tkh,
// tl, tlh, tli, tmh, tn, to, tog, tpi, tr, ts, tsi, tt, tum, tup, tut,
// tvl, tw, ty, tyv, udm, ug, uga, uk, umb, und, ur, uz, vai, ve, vi,
// vo, vot, wa, wak, wal, war, was, wen, wo, xal, xh, yao, yap, yi, yo,
// ypk, za, zap, zbl, zen, zh, znd, zu, zun, zxx, zza”
```

Notes: Available in Mac OS X v10.4 and later.

Note that many of the language codes will not have any supporting locale data in Mac OS X.

12.2.18 lineDirectionForLanguage(isoLangCode as string) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the line direction for the specified ISO language code.

Notes: Returns the line direction for the language. See constants for possible values. If the appropriate direction can't be determined NSLocaleLanguageDirectionUnknown is returned.

Available in Mac OS X v10.6 and later.

12.2.19 localeIdentifier as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: Returns the identifier for the receiver.

Notes: The identifier for the receiver. This may not be the same string that the locale was created with, since NSLocale may canonicalize it.

Available in Mac OS X v10.4 and later.

12.2.20 localeIdentifierFromWindowsLocaleCode(code as Integer) as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a locale identifier from a Windows locale code.

Notes: Available in Mac OS X v10.6 and later.

12.2.21 NSBuddhistCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Buddhist calendar.

Available in Mac OS X v10.4 and later.

12.2.22 NSChineseCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Chinese calendar (unsupported).

Note that the Chinese calendar is not supported in Mac OS X v10.4-10.5. Although you can create a calendar using this constant, the object will not function correctly.

Available in Mac OS X v10.4 and later.

12.2.23 NSGregorianCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Gregorian calendar.

Available in Mac OS X v10.4 and later.

12.2.24 NSHebrewCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Hebrew calendar.

Available in Mac OS X v10.4 and later.

12.2.25 NSIndianCalendar as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Indian calendar

Available in Mac OS X v10.6 and later.

12.2.26 NSIslamicCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Islamic calendar.

Available in Mac OS X v10.4 and later.

12.2.27 NSIslamicCivilCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Islamic civil calendar.

Available in Mac OS X v10.4 and later.

12.2.28 NSISO8601Calendar as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the ISO8601. The ISO8601 calendar is not yet implemented.

Available in Mac OS X v10.6 and later.

12.2.29 NSJapaneseCalendar as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Japanese calendar.

Available in Mac OS X v10.4 and later.

12.2.30 NSLocaleAlternateQuotationBeginDelimiterKey as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

Notes: The key for the alternating begin quotation symbol associated with the locale. In some locales, when quotations are nested, the quotation characters alternate. Thus, NSLocaleQuotationBeginDelimiterKey, then NSLocaleAlternateQuotationBeginDelimiterKey, etc.

The corresponding value is a string.

Available in Mac OS X v10.6 and later.

12.2.31 NSLocaleAlternateQuotationEndDelimiterKey as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

Notes: The key for the alternating end quotation symbol associated with the locale. In some locales, when

quotations are nested, the quotation characters alternate. Thus, `NSLocaleQuotationEndDelimiterKey`, then `NSLocaleAlternateQuotationEndDelimiterKey`, etc.

The corresponding value is a string.

Available in Mac OS X v10.6 and later.

12.2.32 `NSLocaleCalendar` as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

12.2.33 `NSLocaleCollationIdentifier` as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.34 `NSLocaleCollatorIdentifier` as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

Notes: The key for the collation identifier for the locale.

The corresponding value is a string. If unknown, "" is returned.

Available in Mac OS X v10.6 and later.

12.2.35 `NSLocaleCountryCode` as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.36 `NSLocaleCurrencyCode` as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.37 NSLocaleCurrencySymbol as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.38 NSLocaleDecimalSeparator as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.39 NSLocaleExemplarCharacterSet as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.40 NSLocaleGroupingSeparator as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.41 NSLocaleIdentifier as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

Example:

```
dim n as new NSLocaleMBS("de")
```

```
MsgBox n.displayName(n.NSLocaleIdentifier, "en")
```

12.2.42 NSLocaleLanguageCode as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

12.2.43 NSLocaleMeasurementSystem as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

12.2.44 NSLocaleQuotationBeginDelimiterKey as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

Notes: The key for the begin quotation symbol associated with the locale.

The corresponding value is a string.

Available in Mac OS X v10.6 and later.

12.2.45 NSLocaleQuotationEndDelimiterKey as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

Notes: The key for the begin quotation symbol associated with the locale.

The corresponding value is string.

Available in Mac OS X v10.6 and later.

12.2.46 NSLocaleScriptCode as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the displayName and GetString functions.

12.2.47 NSLocaleUsesMetricSystem as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.48 NSLocaleVariantCode as string

Plugin Version: 9.1, Platform: macOS, Targets: All.

Function: One of the constants for the `displayName` and `GetString` functions.

12.2.49 NSPersianCalendar as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Persian calendar.

Available in Mac OS X v10.6 and later.

12.2.50 NSRepublicOfChinaCalendar as string

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: One of the calendar indentifiers.

Notes: Identifier for the Republic of China (Taiwan) calendar.

A Chinese calendar can be created, and one can do calendrical calculations with it, but it should not be used for formatting as the necessary underlying functionality is not functioning correctly yet.

Available in Mac OS X v10.6 and later.

12.2.51 preferredLanguages as string()

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the user's language preference order as an array of strings.

Example:

```
MsgBox join(NSLocaleMBS.preferredLanguages, ", ")
```

```
// shows "de, en, fr, es, it, pt, pt-PT, nl, sv, nb, da, fi, ru, pl, zh-Hans, ja, zh-Hant, ko"
```

Notes: The user's language preference order as an array of String objects, each of which is a canonicalized IETF BCP 47 language identifier.

Available in Mac OS X v10.5 and later.

This is the language the user prefers. To get the country linked to the number and date formats, use the CountryCode property.

12.2.52 systemLocale as NSLocaleMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the "root", canonical locale, that contains fixed "backstop" settings that provide values for otherwise undefined keys.

Notes: Available in Mac OS X v10.4 and later.

12.2.53 windowsLocaleCodeFromLocaleIdentifier(s as string) as Integer

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns a Window locale code from the locale identifier.

Notes: Available in Mac OS X v10.6 and later.

12.2.54 Properties

12.2.55 CollationIdentifier as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The collation associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.CollationIdentifier // shows "" in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.56 CountryCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The country code.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.CountryCode // shows "DE" in Germany
```

Notes: An example value might be "ES".

Available in Mac OS X v10.4 and later.

This code here is the country where the number and date settings belong to. This is not the language the user has. To get the languages, check the preferredLanguages array.
(Read only property)

12.2.57 CurrencyCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The currency code associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.CurrencyCode // shows "EUR" in Germany
```

Notes: Available in Mac OS X v10.4 and later.

(Read only property)

12.2.58 CurrencySymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The currency symbol associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.CurrencySymbol // shows "€" in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.59 DateFull as NSLocaleDateMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The full date format.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.60 DateLong as NSLocaleDateMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The long date format.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.61 DateMedium as NSLocaleDateMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The medium date format.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.62 DateShort as NSLocaleDateMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The short date format.

Example:

```
MsgBox NSLocaleMBS.currentLocale.DateShort.dateFormat  
// in Germany: "dd.MM.yy"
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.63 DecimalSeparator as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The decimal separator associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.DecimalSeparator // shows "," in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.64 GroupingSeparator as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The numeric grouping separator associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.GroupingSeparator // shows "." in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.65 Identifier as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The locale identifier.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.Identifier // shows "de_DE" in Germany
```

Notes: An example value might be "es_ES_PREEURO".
Available in Mac OS X v10.4 and later.
(Read only property)

12.2.66 LanguageCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The language code.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.LanguageCode // shows "de" in Germany
```

Notes: An example value might be "es".
Available in Mac OS X v10.4 and later.
(Read only property)

12.2.67 MeasurementSystem as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The measurement system associated with the locale.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.MeasurementSystem // shows "Metric" in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.68 NumberCurrency as NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The currency style for number formatting.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.69 NumberDecimal as NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Specifies a decimal style for numbers.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.70 NumberPercent as NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The percent style for number formatting.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.71 NumberScientific as NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Specifies a scientific style for numbers.

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.72 NumberSpellOut as NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Specifies a spell-out format; for numbers, "23" becomes "twenty-three".

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.73 ScriptCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The locale script code.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.ScriptCode // shows "" in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.74 UsesMetricSystem as Boolean

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The flag that indicates whether the locale uses the metric system.

Example:

```
dim n as new NSLocaleMBS
```

```
dim v as Variant
```

```
v=n.UsesMetricSystem
```

```
MsgBox v // shows "True" in Germany
```

Notes: Available in Mac OS X v10.4 and later.
(Read only property)

12.2.75 VariantCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The locale variant code.

Example:

```
dim n as new NSLocaleMBS
```

```
MsgBox n.VariantCode // shows "" in Germany
```

Notes: An example value might be "PREEURO".
Available in Mac OS X v10.4 and later.
(Read only property)

12.2.76 Constants

Constants

Constant	Value	Description
NSLocaleLanguageDirectionBottomToTop	4	One of the constants describing the text direction for a language. The language direction is from bottom to top. Available in Mac OS X v10.6 and later.
NSLocaleLanguageDirectionLeftToRight	1	One of the constants describing the text direction for a language. The language direction is from left to right. Available in Mac OS X v10.6 and later.
NSLocaleLanguageDirectionRightToLeft	2	One of the constants describing the text direction for a language. The language direction is from right to left. Available in Mac OS X v10.6 and later.
NSLocaleLanguageDirectionTopToBottom	3	One of the constants describing the text direction for a language. The language direction is from top to bottom. Available in Mac OS X v10.6 and later.
NSLocaleLanguageDirectionUnknown	0	One of the constants describing the text direction for a language. The direction of the language is unknown. Available in Mac OS X v10.6 and later.

12.3 class NSLocaleNumberMBS

12.3.1 class NSLocaleNumberMBS

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: A class for a number format.

Notes: Should be used with the NSLocaleMBS class, but can be used on its own, too.

Blog Entries

- [MBS Xojo Plugins, version 22.2pr1](#)
- [MBS Xojo / Real Studio Plugins, version 13.5pr4](#)
- [MBS Plugins 10.3 Release Notes](#)

12.3.2 Methods

12.3.3 Constructor

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: The constructor.

Notes: Loads values.

No the same values as if you get when using this class with NSLocaleMBS.

See also:

- 12.3.4 Constructor(locale as NSLocaleMBS)

1082

12.3.4 Constructor(locale as NSLocaleMBS)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The constructor.

Example:

```
dim n1 as new NSLocaleNumberMBS
dim n2 as new NSLocaleNumberMBS(NSLocaleMBS.currentLocale)
dim n3 as new NSLocaleNumberMBS(NSLocaleMBS.systemLocale)
```

break // see differences in debugger

Notes: Initializes the object for a given locale.

See also:

- 12.3.3 Constructor

1082

12.3.5 Properties

12.3.6 alwaysShowsDecimalSeparator as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver always shows a decimal separator, even if the number is an integer.

Notes: (Read and Write property)

12.3.7 currencyCode as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the currency code.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.currencycode
// shows for example: e.g "EUR" (in Germany)
```

Notes: A currency code is a three-letter code that is, in most cases, composed of a country's two-character Internet country code plus an extra character to denote the currency unit. For example, the currency code for the Australian dollar is "AUD". Currency codes are based on the ISO 4217 standard.

Available in Mac OS X v10.4 and later.
(Read and Write property)

12.3.8 currencyDecimalSeparator as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the currency decimal separator as a string.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.currencyDecimalSeparator
// shows for example: e.g ",", (in Germany)
```

Notes: (Read and Write property)

12.3.9 currencyGroupingSeparator as String

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The currency grouping separator for the receiver.

Notes: Available in Mac OS X v10.5 and later.

(Read and Write property)

12.3.10 currencySymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the local currency symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.currencycode
// shows for example: e.g "€" (in Germany)
```

Notes: (Read and Write property)

12.3.11 decimalSeparator as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns a string containing the character to represent decimal separators.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.decimalSeparator
// shows for example: e.g "," (in Germany)
```

Notes: (Read and Write property)

12.3.12 exponentSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used as an exponent symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.exponentSymbol
// shows for example: e.g "E" (in Germany)
```

Notes: The exponent symbol is the "E" or "e" in the scientific notation of numbers, as in 1.0e+56.
(Read and Write property)

12.3.13 format as String

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: The format used by the receiver.

Notes: (Read and Write property)

12.3.14 groupingSeparator as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns a string containing the grouping separator.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.groupingSeparator
// shows for example: e.g "." (in Germany)
```

Notes: For example, the grouping separator used in the United States is the comma ("10,000") whereas in France it is the period ("10.000").
(Read and Write property)

12.3.15 hasThousandSeparators as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver's format includes thousand separators.

Notes: (Read and Write property)

12.3.16 internationalCurrencySymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the international currency symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.internationalCurrencySymbol
// shows for example: e.g "EUR" (in Germany)
```

Notes: A country typically has a local currency symbol and an international currency symbol. The local symbol is used within the country, while the international currency symbol is used in international contexts to specify that country's currency unambiguously. The international currency symbol is often represented by a Unicode code point.

(Read and Write property)

12.3.17 Lenient as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver uses heuristics to guess at the date which is intended by a string.

Notes: (Read and Write property)

12.3.18 localizesFormat as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver localizes formats.

Notes: (Read and Write property)

12.3.19 minusSign as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used to represent the minus sign.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.minusSign
// shows for example: e.g "-" (in Germany)
```

Notes: (Read and Write property)

12.3.20 negativeFormat as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the format used to display negative numbers.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.negativeFormat
// shows for example: e.g "#,##0.00 -§" (in Germany)
```

Notes: (Read and Write property)

12.3.21 negativeInfinitySymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the symbol used to represent negative infinity.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.negativeInfinitySymbol
// shows for example: e.g "" (in Germany)
```

Notes: (Read only property)

12.3.22 negativePrefix as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string which is inserted as a prefix to negative values.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.negativePrefix
// shows for example: e.g "-" (in Germany)
```

Notes: (Read and Write property)

12.3.23 negativeSuffix as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string which adds as a suffix to negative values.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.negativeSuffix
// shows for example: e.g ",C" (in Germany)
```

Notes: (Read and Write property)

12.3.24 nilSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used to represent a nil value.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency

MsgBox n.nilSymbol
// shows for example: e.g "" (in Germany)
```

Notes: (Read only property)

12.3.25 notANumberSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the symbol used to represent NaN ("not a number") when it converts values.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency

MsgBox n.notANumberSymbol
// shows for example: e.g "NaN" (in Germany)
```

Notes: (Read only property)

12.3.26 paddingCharacter as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns a string containing the padding character.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency

MsgBox n.paddingCharacter
// shows for example: e.g "*" (in Germany)
```

Notes: (Read and Write property)

12.3.27 PartialStringValidationEnabled as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether partial string validation is enabled.

Notes: (Read and Write property)

12.3.28 percentSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string that is used to represent the percent symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.percentSymbol
// shows for example: e.g. "%" (in Germany)
```

Notes: (Read and Write property)

12.3.29 perMillSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string that is used for the per-thousands symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.perMillSymbol
// shows for example: e.g. ",Ä∞" (in Germany)
```

Notes: (Read and Write property)

12.3.30 plusSign as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used for the plus sign.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.plusSign
// shows for example: e.g "+" (in Germany)
```

Notes: (Read and Write property)

12.3.31 positiveFormat as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the format used to display positive numbers.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.negativeFormat
// shows for example: e.g "#,##0.00 -$" (in Germany)
```

Notes: (Read and Write property)

12.3.32 positiveInfinitySymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used for the positive infinity symbol.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.positiveInfinitySymbol
// shows for example: e.g "" (in Germany)
```

Notes: (Read only property)

12.3.33 positivePrefix as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used as the prefix for positive values.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.positivePrefix
// shows for example: e.g "" (in Germany)
```

Notes: (Read and Write property)

12.3.34 positiveSuffix as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used as the suffix for positive values.

Example:

```
dim l as new NSLocaleMBS
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.positiveSuffix
// shows for example: e.g ",€" (in Germany)
```

Notes: (Read and Write property)

12.3.35 thousandSeparator as String

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver's format includes thousand separators.

Notes: (Read and Write property)

12.3.36 usesGroupingSeparator as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver uses the grouping separator.

Notes: (Read and Write property)

12.3.37 usesSignificantDigits as Boolean

Plugin Version: 10.3, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the receiver uses significant digits.

Notes: (Read and Write property)

12.3.38 zeroSymbol as String

Plugin Version: 7.4, Platform: macOS, Targets: All.

Function: Returns the string used as the zero symbol.

Example:

```
dim l as new NSLocaleMBS
```

```
dim n as NSLocaleNumberMBS = l.NumberCurrency
```

```
MsgBox n.zeroSymbol
```

```
// shows for example: e.g "0" (in Germany)
```

Notes: (Read and Write property)

Chapter 13

DiscRecording

13.1 class DRNotificationCenterMBS

13.1.1 class DRNotificationCenterMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop only.

Function: A class for notifications.

Notes: All methods in this class will catch exceptions from Cocoa and raise a NSErrorMBS instead. Using the message, name and reason properties you can see what was the reason for this exception. Please report if you find a method which does not handle exceptions correct.

13.1.2 Methods

13.1.3 addObserver(observer as NSNotificationCenterObserverMBS, name as string="", theObject as Variant=nil)

Plugin Version: 10.4, Platform: macOS, Targets: Desktop only.

Function: Adds an entry to the receiver's dispatch table with an observer and optional criteria: notification name and sender.

Notes: observer: Object registering as an observer. This value must not be nil.

name: The name of the notification for which to register the observer; that is, only notifications with this name are delivered to the observer. If you pass nil, the notification center doesn't use a notification's name to decide whether to deliver it to the observer.

theObject: The object whose notifications the observer wants to receive; that is, only notifications sent by this sender are delivered to the observer. If you pass nil, the notification center doesn't use a notification's sender to decide whether to deliver it to the observer.

13.1.4 Constructor

Plugin Version: 10.4, Platform: macOS, Targets: Desktop only.

Function: The constructor to create a new instance of notification center pointing to the default notification center.

Notes: A DRNotificationCenter object (or simply, notification center) is essentially a notification dispatch table. It notifies all observers of notifications meeting specific criteria. This information is encapsulated in NSNotification objects, also known as notifications. Client objects register themselves with the notification center as observers of specific notifications posted by DiscRecording. When an event occurs, DiscRecording posts an appropriate notification to the notification center. The notification center dispatches a message to each registered observer, passing the notification as the sole argument.

There are two main differences between a DRNotificationCenter and the NSNotificationCenter from Foundation. First is that only Disc Recording posts notifications received through this mechanism. You use this to obtain device plug/unplug events, burn status, etc. Second, there can be multiple notification centers active at once. Each run loop of your application will have its own notification center and notifications from that notification center will be posted to the runloop it was created on.

13.1.5 removeObserver(observer as NSNotificationObserverMBS, name as string, theObject as Variant=nil)

Plugin Version: 10.4, Platform: macOS, Targets: Desktop only.

Function: Removes matching entries from the receiver's dispatch table.

Notes: notificationObserver: Observer to remove from the dispatch table. Specify an observer to remove only entries for this observer. Must not be nil, or message will have no effect.

notificationName: Name of the notification to remove from dispatch table. Specify a notification name to remove only entries that specify this notification name. When nil, the receiver does not use notification names as criteria for removal.

notificationSender: Sender to remove from the dispatch table. Specify a notification sender to remove only entries that specify this sender. When nil, the receiver does not use notification senders as criteria for removal.

Be sure to invoke this method before the observer object or any object specified in addObserver is deallocated.

Chapter 14

Notifications

14.1 class NSDistributedNotificationCenterMBS

14.1.1 class NSDistributedNotificationCenterMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: The NSDistributedNotificationCenter class provides a way to send notifications to objects in other tasks.

Notes: It takes NSNotification objects and broadcasts them to any objects in other tasks that have registered for the notification with their task's default distributed notification center.

The NSDistributedNotificationCenter class implements a notification center that can distribute notifications asynchronously to tasks other than the one in which the notification was posted. An instance of this class are known as a distributed notification center.

Each task has a default distributed notification center that you access with the defaultCenter class method. There may be different types of distributed notification centers. Currently there is a single type—NSLocalNotificationCenterType. This type of distributed notification center handles notifications that can be sent between tasks on a single computer. For communication between tasks on different computers, use Distributed Objects Programming Topics.

Posting a distributed notification is an expensive operation. The notification gets sent to a system-wide server that distributes it to all the tasks that have objects registered for distributed notifications. The latency between posting the notification and the notification's arrival in another task is unbounded. In fact, when too many notifications are posted and the server's queue fills up, notifications may be dropped.

Subclass of the NSNotificationCenterMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.3pr11](#)

- [MBS REALbasic Plugins, version 10.6pr6](#)

14.1.2 Methods

14.1.3 addObserver(observer as NSNotificationObserverMBS, name as string, theObject as Variant, suspensionBehavior as Integer)

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Adds an entry to the receiver's dispatch table with a specific observer and suspended-notifications behavior, and optional notification name and sender.

Notes: observer: Object registering as an observer. Must not be nil.

name: The name of the notification for which to register the observer; that is, only notifications with this name are delivered to the observer. When nil, the notification center doesn't use a notification's name to decide whether to deliver it to the observer.

theObject: The object whose notifications the observer wants to receive; that is, only notifications sent by this sender are delivered to the observer. When nil, the notification center doesn't use a notification's sender to decide whether to deliver it to the observer.

suspensionBehavior: Notification posting behavior when notification delivery is suspended.

The receiver does not retain notificationObserver. Therefore, you should always send removeObserver to the receiver before releasing notificationObserver.

14.1.4 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new instance with the default distributed notification center, representing the local notification center for the computer.

Notes: Default distributed notification center for the computer.

This method calls notificationCenterForType: with an argument of NSLocalNotificationCenterType.

14.1.5 defaultCenter as NSDistributedNotificationCenterMBS

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the default distributed notification center, representing the local notification center for the computer.

Notes: Default distributed notification center for the computer.

This method calls `notificationCenterForType:` with an argument of `NSLocalNotificationCenterType`.

14.1.6 `notificationCenterForType(name as string)` as `NSDistributedNotificationCenterMBS`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the distributed notification center for a particular notification center type.

Notes: `name`: Notification center type being inquired about.

Currently only one type, `NSLocalNotificationCenterType`, is supported.

14.1.7 `NSLocalNotificationCenterType` as string

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: This constant specifies the notification center type.

Notes: Distributes notifications to all tasks on the sender's computer.

14.1.8 `postNotificationName(name as string, theObject as string, userInfo as dictionary, deliverImmediately as boolean)`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a notification with information and an immediate-delivery specifier, and posts it to the receiver.

Notes: `name`: Name of the notification to post. Must not be nil.

`theObject`: Sender of the notification. May be "".

`userInfo`: Dictionary containing additional information. May be nil.

`deliverImmediately`: Specifies when to deliver the notification. When false, the receiver delivers notifications to their observers according to the suspended-notification behavior specified in the corresponding dispatch table entry. When true, the receiver delivers the notification immediately to its observers.

This is the preferred method for posting notifications.

The `notificationInfo` dictionary is serialized as a property list, so it can be passed to another task. In the receiving task, it is deserialized back into a dictionary. This serialization imposes some restrictions on the objects that can be placed in the `notificationInfo` dictionary. See XML Property Lists for details.

See also:

- 14.1.9 `postNotificationName(name as string, theObject as string, userInfo as dictionary, options as UInt32)` 1100

14.1.9 `postNotificationName(name as string, theObject as string, userInfo as dictionary, options as UInt32)`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a notification with information, and posts it to the receiver.

Example:

```
dim notifyData as string = "Hello World"
dim notifyName as string = "Test"
```

```
Dim notifyDict As New Dictionary
notifyDict.Value("nData")=notifyData
NSDistributedNotificationCenterMBS.DefaultCenter.postNotificationName(notifyName, "", notifyDict, 2)
```

Notes: name: Name of the notification to post. Must not be nil.

theObject: Sender of the notification. May be "".

userInfo: Dictionary containing additional information. May be nil.

options: Specifies how the notification is posted to the task and when to deliver it to its observers.

The userInfo dictionary is serialized as a property list, so it can be passed to another task. In the receiving task, it is deserialized back into a dictionary. This serialization imposes some restrictions on the objects that can be placed in the notificationInfo dictionary. See XML Property Lists for details.

See also:

- 14.1.8 `postNotificationName(name as string, theObject as string, userInfo as dictionary, deliverImmediately as boolean)` 1099

14.1.10 Properties

14.1.11 `suspended as boolean`

Plugin Version: 9.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A boolean value that indicates whether notification delivery is suspended.

Notes: The `NSApplication` class automatically suspends distributed notification delivery when the application is not active. Applications based on the Application Kit framework should let AppKit manage the suspension of notification delivery. Foundation-only programs may have occasional need to use this method. (Read and Write computed property)

14.1.12 Constants

Constants

Constant	Value	Description
NSNotificationDeliverImmediately	1	One of the option constants to specify the behavior of notification delivery for the <code>postNotificationName</code> method. When set, the notification is delivered immediately to all notification observers of their suspension behavior or suspension state. When not set, the notification follows the normal suspension behavior of notification observers to the <code>postNotificationName</code> method.
NSNotificationPostToAllSessions	2	One of the option constants to specify the behavior of notification delivery for the <code>postNotificationName</code> method. When set, the notification is posted to all sessions. When not set, the notification is sent only to applications within the same login session.
NSNotificationSuspensionBehaviorCoalesce	2	One of the constants to specify the types of notification suspension behaviors. The server only queues the last notification of the specified type. All earlier notifications are dropped. In cover methods for <code>NSNotificationSuspensionBehavior</code> is not an explicit argument, <code>NSNotificationSuspensionBehaviorCoalesce</code> is the default.
NSNotificationSuspensionBehaviorDeliverImmediately	4	One of the constants to specify the types of notification suspension behaviors. The server delivers notifications matching this registration immediately, whether <code>Suspended</code> is set to true. When a notification matching this behavior is matched, it has the effect of first flushing any pending notifications. The effect is as if <code>setSuspended:</code> with an argument of false, the application is suspended, followed by the notification being delivered, followed by a transition back to the previous suspension state.
NSNotificationSuspensionBehaviorDrop	1	One of the constants to specify the types of notification suspension behaviors. The server does not queue any notifications with this registration. <code>Suspended</code> is set to false.
NSNotificationSuspensionBehaviorHold	3	One of the constants to specify the types of notification suspension behaviors. The server holds all matching notifications until the queue is full (queue size determined by the server), at which point the server delivers the notifications.

14.2 class NotificationCenterMBS

14.2.1 class NotificationCenterMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: An NotificationCenter object (or simply, notification center) provides a mechanism for broadcasting information within a task.

Notes: An NotificationCenter object is essentially a notification dispatch table.

This is for sending broadcast messages from one application to other application on same computer. For the notification center in Mac OS X 10.8, please use NSUserNotificationMBS class.

Blog Entries

- [MBS Real Studio Plugins, version 12.3pr11](#)

14.2.2 Methods

14.2.3 addObserver(observer as NotificationCenterObserverMBS, name as string="", theObject as Variant=nil)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Adds an entry to the receiver's dispatch table with an observer and optional criteria: notification name and sender.

Notes: observer: Object registering as an observer. This value must not be nil.

name: The name of the notification for which to register the observer; that is, only notifications with this name are delivered to the observer. If you pass nil, the notification center doesn't use a notification's name to decide whether to deliver it to the observer.

theObject: The object whose notifications the observer wants to receive; that is, only notifications sent by this sender are delivered to the observer. If you pass nil, the notification center doesn't use a notification's sender to decide whether to deliver it to the observer.

14.2.4 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The constructor to create a new instance of notification center pointing to the default notification center.

Notes: The current task's default notification center, which is used for system notifications.

14.2.5 defaultCenter as NSNotificationCenterMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the task's default notification center.

Notes: The current task's default notification center, which is used for system notifications.

14.2.6 postNotification(notification as NSNotificationMBS)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Posts a given notification to the receiver.

Notes: notification: The notification to post. This value must not be nil.

14.2.7 postNotificationName(name as string)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Creates a notification with a given name and sender and posts it to the receiver.

Notes: name: The name of the notification.

theObject: The object posting the notification.

See also:

- 14.2.8 postNotificationName(name as string, theObject as Variant) 1103
- 14.2.9 postNotificationName(name as string, theObject as Variant, userInfo as dictionary) 1103

14.2.8 postNotificationName(name as string, theObject as Variant)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Creates a notification with a given name and sender and posts it to the receiver.

Notes: name: The name of the notification.

theObject: The object posting the notification.

See also:

- 14.2.7 postNotificationName(name as string) 1103
- 14.2.9 postNotificationName(name as string, theObject as Variant, userInfo as dictionary) 1103

14.2.9 postNotificationName(name as string, theObject as Variant, userInfo as dictionary)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Creates a notification with a given name, sender, and information and posts it to the receiver.

Notes: name: The name of the notification.

theObject: The object posting the notification.

userInfo: Information about the the notification. May be nil.

This method is the preferred method for posting notifications.

See also:

- 14.2.7 postNotificationName(name as string) 1103
- 14.2.8 postNotificationName(name as string, theObject as Variant) 1103

14.2.10 removeObserver(observer as NSNotificationObserverMBS)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Removes all the entries specifying a given observer from the receiver's dispatch table.

Notes: observer: The observer to remove. Must not be nil.

Be sure to invoke this method before notificationObserver or any object specified in addObserver is deallocated.

See also:

- 14.2.11 removeObserver(observer as NSNotificationObserverMBS, name as string, theObject as Variant=nil) 1104

14.2.11 removeObserver(observer as NSNotificationObserverMBS, name as string, theObject as Variant=nil)

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Removes matching entries from the receiver's dispatch table.

Notes: notificationObserver: Observer to remove from the dispatch table. Specify an observer to remove only entries for this observer. Must not be nil, or message will have no effect.

notificationName: Name of the notification to remove from dispatch table. Specify a notification name to remove only entries that specify this notification name. When nil, the receiver does not use notification names as criteria for removal.

notificationSender: Sender to remove from the dispatch table. Specify a notification sender to remove only entries that specify this sender. When nil, the receiver does not use notification senders as criteria for removal.

Be sure to invoke this method before the observer object or any object specified in addObserver is deallocated.

See also:

- 14.2.10 removeObserver(observer as NSNotificationObserverMBS) 1104

14.2.12 Properties

14.2.13 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal reference to the notification center.

Notes: (Read and Write property)

14.3 class NSNotificationMBS

14.3.1 class NSNotificationMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: NSNotification objects encapsulate information so that it can be broadcast to other objects by an NSNotificationCenter object.

Notes: An NSNotification object (referred to as a notification) contains a name, an object, and an optional dictionary. The name is a tag identifying the notification. The object is any object that the poster of the notification wants to send to observers of that notification (typically, it is the object that posted the notification). The dictionary stores other related objects, if any. NSNotification objects are immutable objects.

You can create a notification object with the class method `notificationWithName`. However, you don't usually create your own notifications directly. The `NSNotificationCenter` method `postNotificationName` allow you to conveniently post a notification without creating it first.

On Windows you can use `WinNotificationMBS` class for similar functionality.

This is for sending broadcast messages from one application to other application on same computer. For the notification center in Mac OS X 10.8, please use `NSUserNotificationMBS` class.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.3pr4](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr3](#)
- [MBS Real Studio Plugins, version 12.1pr5](#)
- [Windows Notifications](#)

14.3.2 Methods

14.3.3 Constructor(handle as Integer)

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Creates a notification object with a NSNotification reference.

Notes: The object is retained and later in destructor it is released.

See also:

- 14.3.4 `Constructor(name as string, theObject as Variant = nil, userInfo as dictionary = nil)` 1107

14.3.4 Constructor(name as string, theObject as Variant = nil, userInfo as dictionary = nil)

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates a notification object with a specified name, object, and user information.

Example:

```
dim n as new NSNotificationMBS("Hello")
MsgBox n.name
```

Notes: name: The name for the new notification. May not be nil.

theObject: The object for the new notification. Can be nil.

userInfo: The user information dictionary for the new notification. May be nil.

See also:

- 14.3.3 Constructor(handle as Integer)

1106

14.3.5 notificationWithName(name as string, theObject as Variant = nil, userInfo as dictionary = nil) as NSNotificationMBS

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Returns a notification object with a specified name, object, and user information.

Example:

```
dim n as NSNotificationMBS = NSNotificationMBS.notificationWithName("Hello")
MsgBox n.name
```

Notes: name: The name for the new notification. May not be nil.

theObject: The object for the new notification. Can be nil.

userInfo: The user information dictionary for the new notification. May be nil.

14.3.6 Print

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Writes description for this event descriptor to the console.

Notes: You can see result in Console.app.

14.3.7 Properties

14.3.8 description as string

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: The descriptor for this notification.

Example:

```
dim n as NSNotificationMBS = NSNotificationMBS.notificationWithName("Hello")
MsgBox n.description
```

Notes: This is a text representation for debugging.
(Read only property)

14.3.9 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal reference to the NSNotification object.

Notes: (Read and Write property)

14.3.10 name as string

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the name of the notification.

Notes: The name of the notification. Typically you use this method to find out what kind of notification you are dealing with when you receive a notification.

Notification names can be any string. To avoid name collisions, you might want to use a prefix that's specific to your application.

(Read only property)

14.3.11 objectHandle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The reference to the object.

Notes: May be useful for declares.

(Read only property)

14.3.12 objectVariant as Variant

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the object associated with the notification.

Notes: The object associated with the notification. This is often the object that posted this notification. It may be nil.

Typically you use this method to find out what object a notification applies to when you receive a notification.

See the FAQ for the list of supported NSObject types for variant conversion.
(Read only property)

14.3.13 userInfo as dictionary

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: Returns the user information dictionary associated with the receiver.

Example:

```
// tests dictionary conversion:
```

```
dim d as new Dictionary
```

```
dim x as new Dictionary
x.Value("Hello") = "World"
```

```
dim b as Boolean = true
dim s as Single = 2.3
dim dd as Double = 3.4
dim h as int64 = 1234
```

```
d.Value(1) = 1
d.Value(2) = 2.0
d.Value(3) = x
d.Value(4) = "Hello"
d.Value(5) = b
d.Value(6) = s
d.Value(7) = dd
d.Value(8) = h
```

```
dim n as new NSNotificationMBS("test", nil, d)
```

```
dim nd as Dictionary = n.userInfo
```

```
// check nd object
break
```

Notes: Returns the user information dictionary associated with the receiver. May be nil.
The user information dictionary stores any additional objects that objects receiving the notification might use.
(Read only property)

14.4 class NSNotificationObserverMBS

14.4.1 class NSNotificationObserverMBS

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The notification class which you use to receive notifications.

Notes: To use this class, please create a subclass. There you can put code in the event handler to do whatever you need in case your notification is received.

Then you create in code objects from this class. Normally only one. And you call addObserver on the NSNotificationCenterMBS or NSDistributedNotificationCenterMBS objects you have to tell the system which notifications you want to receive.

Do not forget to call removeObserver on the notification center for all your observers to avoid crashes.

Blog Entries

- [Tip of the day: Windows Notifications](#)
- [MBS Real Studio Plugins, version 12.3pr11](#)

14.4.2 Methods

14.4.3 Constructor

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The constructor which creates the observer.

Notes: On success the handle property is not zero.

14.4.4 Destructor

Plugin Version: 12.3, Platform: macOS, Targets: All.

Function: The Destructor.

Notes: Removes in plugin version 12.3 and newer the observer from the NSNotificationCenterMBS and NSDistributedNotificationCenterMBS to avoid crashes.

14.4.5 Properties

14.4.6 Handle as Integer

Plugin Version: 9.7, Platform: macOS, Targets: All.

Function: The internal handle to the observer object.

Notes: (Read and Write property)

14.4.7 Events

14.4.8 GotNotification(notification as NSNotificationMBS)

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

Function: The event called if a new notification is received.

Notes: The MBS Plugin makes sure you receive this event always on the main thread.

Chapter 15

Process

15.1 class NSProcessInfoActivityMBS

15.1.1 class NSProcessInfoActivityMBS

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The class for an activity.

Example:

```
dim Activity as NSProcessInfoActivityMBS // property in your window, control, thread, app
dim AllowAppNap as boolean // allow or not?

dim ProcessInfo as NSProcessInfoMBS = NSProcessInfoMBS.processInfo
if AllowAppNap then
    Activity = nil
else
    // disable sleep to let us make something...
    Activity = ProcessInfo.beginActivity(NSProcessInfoMBS.NSActivityBackground, "Backup running")
end if
```

Notes: For Windows, please check WindowsThreadExecutionStateMBS to prevent system sleep. This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

15.1.2 Methods

15.1.3 Constructor

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Private constructor.

15.1.4 Destructor

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The destructor.

Notes: If you missed to call `endActivity`, the destructor will do it.

15.1.5 Properties

15.1.6 Handle as Integer

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

15.1.7 Options as Integer

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The options used to create activity.

Notes: (Read and Write property)

15.1.8 Reason as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The reason used to create the activity.

Notes: (Read and Write property)

15.2 class NSProcessInfoMBS

15.2.1 class NSProcessInfoMBS

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: The NSProcessInfo class provides methods to access information about the current process.

Notes: Each process has a single, shared NSProcessInfo object, known as process information agent.

The process information agent can return such information as the arguments, environment variables, host name, or process name. The processInfo class method returns the shared agent for the current process—that is, the process whose object sent the message. For example, the following line returns the NSProcessInfo object, which then provides the name of the current process:

```
dim processInfo as new NSProcessInfoMBS
dim processName as string = processInfo.processName
```

The NSProcessInfo class also includes the operatingSystem method, which returns an enum constant identifying the operating system on which the process is executing.

NSProcessInfo objects attempt to interpret environment variables and command-line arguments in the user's default C string encoding if they cannot be converted to Unicode as UTF-8 strings. If neither conversion works, these values are ignored by the NSProcessInfo object.

For Windows, please check WindowsThreadExecutionStateMBS to prevent system sleep.

Blog Entries

- [MBS Xojo Plugins, version 24.1pr1](#)
- [MBS Xojo Plugins, version 23.5pr5](#)
- [News from the MBS Xojo Plugins Version 21.3](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 21.3](#)
- [MBS Xojo Plugins, version 21.3pr2](#)
- [MBS Xojo Plugins Version 21.0 News](#)
- [MBS Xojo Plugins, version 21.0pr6](#)
- [MBS Xojo Plugins, version 18.5pr3](#)
- [MBS Xojo / Real Studio Plugins, version 16.4pr6](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.4](#)

Xojo Developer Magazine

- [19.5, page 9: News](#)

15.2.2 Methods

15.2.3 `argument(index as Integer) as string`

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns the command-line argument for the process with the given index.

Example:

```
dim p as new NSProcessInfoMBS
```

```
dim i,c as Integer
```

```
c=p.argumentCount-1
```

```
for i=0 to c
```

```
MsgBox p.argument(i)
```

```
next
```

15.2.4 `arguments as string()`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns the command-line arguments for the process.

Notes: Returns array of strings with the process's command-line arguments.

15.2.5 `beginActivity(options as Integer, reason as string) as NSProcessInfoActivityMBS`

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Begin an activity using the given options and reason.

Example:

```
dim Activity as NSProcessInfoActivityMBS // property in your window, control, thread, app
```

```
dim AllowAppNap as boolean // allow or not?
```

```
dim ProcessInfo as NSProcessInfoMBS = NSProcessInfoMBS.processInfo
```

```
if AllowAppNap then
```

```
Activity = nil
```

```
else
```

```
// disable sleep to let us make something...
```

```
Activity = ProcessInfo.beginActivity(NSProcessInfoMBS.NSActivityBackground, "Backup running")
end if
```

Notes: options: Options for the activity. See constants for possible values.
reason: A string used in debugging to indicate the reason the activity began.

Returns an object token representing the activity.

Indicate completion of the activity by calling `endActivity` passing the returned object as the argument.
Available in OS X v10.9 and later.

For Windows, please check `WindowsThreadExecutionStateMBS` to prevent system sleep.

15.2.6 Constructor

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: The constructor.

15.2.7 `disableAutomaticTermination(Reason as string)`

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Decrement the counter tracking the number of automatic quit opt-out requests.

Notes: When this counter is greater than zero, the app will be considered 'active' and ineligible for automatic termination.

An example of using this would be disabling autoquitting when the user of an instant messaging application signs on, due to it requiring a background connection to be maintained even if the app is otherwise inactive. Each pair of calls should have a matching "reason" argument, which can be used to easily track why an application is or is not automatically terminable.

A given reason can be used more than once at the same time (for example: two files are transferring over the network, each one disables automatic termination with the reason "file transfer in progress")

15.2.8 `disableSuddenTermination`

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Disables the application for quickly killing using sudden termination.

Notes: This method increments the sudden termination counter. When the termination counter reaches 0

the application allows sudden termination.

By default the sudden termination counter is set to 1. This can be overridden in your application Info.plist. See "Sudden Termination" for more information and debugging suggestions.

Available in Mac OS X v10.6 and later.

15.2.9 enableAutomaticTermination(Reason as string)

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Increment the counter tracking the number of automatic quit opt-out requests.

Notes: When this counter is greater than zero, the app will be considered 'active' and ineligible for automatic termination.

An example of using this would be disabling autoquitting when the user of an instant messaging application signs on, due to it requiring a background connection to be maintained even if the app is otherwise inactive. Each pair of calls should have a matching "reason" argument, which can be used to easily track why an application is or is not automatically terminable.

A given reason can be used more than once at the same time (for example: two files are transferring over the network, each one disables automatic termination with the reason "file transfer in progress")

15.2.10 enableSuddenTermination

Plugin Version: 9.6, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables the application for quick killing using sudden termination.

Notes: This method decrements the sudden termination counter. When the termination counter reaches 0 the application allows sudden termination.

By default the sudden termination counter is set to 1. This can be overridden in your application Info.plist. See "Sudden Termination" for more information and debugging suggestions.

Available in Mac OS X v10.6 and later.

15.2.11 endActivity(activity as NSProcessInfoActivityMBS)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Ends the given activity.

Notes: activity: An activity object returned by beginActivity.

Available in OS X v10.9 and later.

15.2.12 NSActivityLatencyCritical as UInt64

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: One of the activity option constants.

Notes: Flag to indicate the activity requires the highest amount of timer and I/O precision available.

Important: Very few applications should need to use this constant.

Available in OS X v10.9 and later.

Value is &hFF00000000.

15.2.13 NSProcessInfoPowerStateDidChangeNotification as String

Plugin Version: 24.1, Platform: macOS, Targets: All.

Function: The name of the notification that is posted when the power state of a device changes.

Notes: After your observer receives this notification, query the `lowPowerModeEnabled` property to determine the current power state of the device. If Low Power Mode is active, take appropriate steps to reduce activity in your app. Otherwise, your app can resume normal operations.

The notification object is an `NSProcessInfo` instance.

15.2.14 NSProcessInfoThermalStateDidChangeNotification as String

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: The notification name to use with `NSNotificationObserverMBS`.

Notes: This notification is posted once the thermal state of the system has changed. Once the notification is posted, use the `thermalState` property to retrieve the current thermal state of the system.

You can use this opportunity to take corrective action in your application to help cool the system down. Work that could be done in the background or at opportunistic times should be using the Quality of Service levels in `NSOperation` or the `NSBackgroundActivityScheduler` API.

This notification is posted on the global dispatch queue. Register for it using the default notification center. The object associated with the notification is `NSProcessInfoMBS.processInfo`.

15.2.15 `operationSystemVersion`(byref Major as Integer, byref Minor as Integer, byref Patch as Integer)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Queries operation system version.

Notes: One call to get all three parts.

15.2.16 `processInfo` as `NSProcessInfoMBS`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns the process information agent for the process.

15.2.17 Properties

15.2.18 `activeProcessorCount` as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Provides the number of active processing cores available on the computer.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox str(p.activeProcessorCount)
```

Notes: Available in Mac OS X v10.5 and later.

(Read only property)

15.2.19 `argumentsCount` as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: The number of the command-line arguments.

Example:

```
dim p as new NSProcessInfoMBS
```

```
dim i,c as Integer
```

```
c=p.argumentCount-1
```

```

for i=0 to c
MsgBox p.argument(i)
next

```

Notes: (Read only property)

15.2.20 automaticTerminationSupportEnabled as boolean

Plugin Version: 11.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Marks the calling app as supporting automatic termination.

Notes: Without calling this or setting the equivalent Info.plist key (NSSupportsAutomaticTermination), the above methods (disableAutomaticTermination/enableAutomaticTermination) have no effect, although the counter tracking automatic termination opt-outs is still kept up to date to ensure correctness if this is called later. Currently, passing false has no effect.

This should be called during applicationDidFinishLaunching or earlier.

(Read and Write property)

15.2.21 environment as dictionary

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns the variable names and their values in the environment from which the process was launched.

Example:

```

dim p as new NSProcessInfoMBS
dim d as Dictionary = p.environment
MsgBox str(d.Count)

```

Notes: (Read only property)

15.2.22 globallyUniqueString as string

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns a global unique identifier for the process.

Example:

```

dim p as new NSProcessInfoMBS

```

`MsgBox p.globallyUniqueString // shows for example "072EC09A-4825-11DD-BDC0-001D4F46F5E0-18405-00000CA853EB5B46"`

Notes: Returns the Global ID for the process. The ID includes the host name, process ID, and a time stamp, which ensures that the ID is unique for the network.

This method generates a new string each time it is invoked, so it also uses a counter to guarantee that strings created from the same process are unique.

(Read only property)

15.2.23 Handle as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: The internal reference to the `NSProcessInfo` object.

Notes: (Read and Write property)

15.2.24 hostName as string

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns the name of the host computer.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox p.hostName // for example "iMac.local"
```

Notes: (Read only property)

15.2.25 isiOSAppOnMac as Boolean

Plugin Version: 21.0, Platform: macOS, Targets: All.

Function: A Boolean value that indicates whether the process is an iPhone or iPad app running on a Mac.

Notes: The value of this property is true only when the process is an iOS app running on a Mac. The value of the property is false for all other apps on the Mac, including Mac apps built using Mac Catalyst. The property is also false for processes running on platforms other than macOS.

(Read only property)

15.2.26 isLowPowerModeEnabled as Boolean

Plugin Version: 21.3, Platforms: macOS, iOS, Targets: All.

Function: Retrieve the current setting of the system for the low power mode setting.

Notes: On systems where the low power mode is unknown or unsupported, the value returned from the lowPowerModeEnabled property is always false.

Returns true if low power mode is enabled for macOS 12 or iOS 9 and newer.

(Read only property)

15.2.27 operatingSystem as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns a constant to indicate the operating system on which the process is executing.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox str(p.operatingSystem) // shows 5 = NSMACHOperatingSystem
```

Notes: Operating system identifier. See "Constants" for a list of possible values. In Mac OS X, it's NSMACHOperatingSystem.

(Read only property)

15.2.28 operatingSystemName as string

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns a string containing the name of the operating system on which the process is executing.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox p.operatingSystemName // shows "NSMACHOperatingSystem"
```

Notes: Operating system name. In Mac OS X, it's "NSMACHOperatingSystem"

(Read only property)

15.2.29 `operatingSystemVersionString` as string

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns a string containing the version of the operating system on which the process is executing.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox p.operatingSystemVersionString // "Version 10.5.4 (Build 9E17)"
```

Notes: Returns the Operating system version. This string is human readable, localized, and is appropriate for displaying to the user. This string is not appropriate for parsing.
(Read only property)

15.2.30 `operationSystemVersionMajor` as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Queries major OS version.

Notes: (Read only property)

15.2.31 `operationSystemVersionMinor` as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Queries minor OS version.

Notes: (Read only property)

15.2.32 `operationSystemVersionPatch` as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Queries OS version patch level.

Notes: (Read only property)

15.2.33 `physicalMemory` as UInt64

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Provides the amount of physical memory on the computer.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox str(p.physicalMemory)
```

Notes: Available in Mac OS X v10.5 and later.
(Read only property)

15.2.34 processIdentifier as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Returns the identifier of the process.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox str(p.processIdentifier)
```

Notes: (Read only property)

15.2.35 processName as string

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: The name of the process.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox p.processName
```

Notes: The process name is used to register application defaults and is used in error messages. It does not uniquely identify the process.

You can assign a new value, but:

User defaults and other aspects of the environment might depend on the process name, so be very careful if you change it. Setting the process name in this manner is not thread safe.

(Read and Write property)

15.2.36 processorCount as Integer

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Provides the number of processing cores available on the computer.

Example:

```
dim p as new NSProcessInfoMBS
```

```
MsgBox str(p.processorCount)
```

Notes: Available in Mac OS X v10.5 and later.

(Read only property)

15.2.37 systemUptime as Double

Plugin Version: 9.6, Platform: macOS, Targets: All.

Function: Returns the how long it has been since the computer has been restarted.

Notes: Returns an NSTimeInterval indicating how long system the computer has been restarted.

Available in Mac OS X v10.6 and later.

(Read only property)

15.2.38 thermalState as Integer

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Retrieve the current thermal state of the system.

Example:

```
dim n as NSProcessInfoMBS = NSProcessInfoMBS.processInfo
```

```
Select case n.thermalState
case n.NSProcessInfoThermalStateNominal
MsgBox "Thermal State: Nominal"
case n.NSProcessInfoThermalStateFair
MsgBox "Thermal State: Fair"
case n.NSProcessInfoThermalStateSerious
MsgBox "Thermal State: Serious"
case n.NSProcessInfoThermalStateCritical
```

```

MsgBox "Thermal State: Critical"
else
MsgBox "Thermal State: Unknown"
end Select

```

Notes: On systems where thermal state is unknown or unsupported, the value returned from the thermal-State property is always NSProcessInfoThermalStateNominal.

Available in Mac OS X 10.10.3 and newer.

Returns -1 if function is called on older Mac OS X versions, Linux or Windows.

(Read only property)

15.2.39 Constants

Constants

Constant	Value	Description
NSHPUXOperatingSystem	4	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the HP UX operating system.
NSMACHOperatingSystem	5	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the Mac OS X operating system.
NSOSF1OperatingSystem	7	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the OSF/1 operating system.
NSSolarisOperatingSystem	3	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the Solaris operating system.
NSSunOSOperatingSystem	6	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the Sun OS operating system.
NSWindows95OperatingSystem	2	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the Windows 95 operating system.
NSWindowsNTOperatingSystem	1	One of the following constants are provided by the NSProcessInfo class as return values for operatingSystem. Indicates the Windows NT operating system.

Activity Options

Constant	Value	Description
<code>NSActivityAutomaticTerminationDisabled</code>	<code>&h8000</code>	Flag to prevent automatic termination. Available in OS X v10.9 and later.
<code>NSActivityBackground</code>	<code>&h000000FF</code>	Flag to indicate the app has initiated some kind of result of user request. Available in OS X v10.9 and later.
<code>NSActivityIdleDisplaySleepDisabled</code>	<code>&h1000000000</code>	Flag to require the screen to stay powered on. Available in OS X v10.9 and later.
<code>NSActivityIdleSystemSleepDisabled</code>	<code>&h100000</code>	Flag to prevent idle sleep. This is included in <code>NSActivityUserInitiatedAllowingIdleSystemSleep</code> . Available in OS X v10.9 and later.
<code>NSActivitySuddenTerminationDisabled</code>	<code>&h4000</code>	Flag to prevent sudden termination. This is included in <code>NSActivityUserInitiatedAllowingSuddenTerminationDisabled</code> . Available in OS X v10.9 and later.
<code>NSActivityUserInitiated</code>	<code>&h00FFFFFF</code>	Flag to indicate the app is performing a user-requested task. Available in OS X v10.9 and later.
<code>NSActivityUserInitiatedAllowingIdleSystemSleep</code>	<code>&h00EFFFFFF</code>	Flag to indicate the app is performing a user-requested task and the system can sleep on idle. Available in OS X v10.9 and later.

Thermal State Constants

Constant	Value	Description
<code>NSProcessInfoThermalStateCritical</code>	3	System performance is significantly impacted and the Mac needs to cool down. Recommendation: reduce application's usage of CPU, GPU, and I/O to the minimum level needed to respond to user actions. Consider stopping use of camera and other peripherals if your application is using them.
<code>NSProcessInfoThermalStateFair</code>	1	The system has reached a state where fans may become audible.
<code>NSProcessInfoThermalStateNominal</code>	0	No corrective action is needed.
<code>NSProcessInfoThermalStateSerious</code>	2	Fans are running at maximum speed, system performance maybe impacted. Recommendation: reduce application's usage of CPU, GPU and I/O, if possible. Switch to lower quality visual effects, reduce frame rates.

Chapter 16

Window

16.1 class DesktopWindow

16.1.1 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.

16.1.2 Methods

16.1.3 NSPanelMBS as NSPanelMBS

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

Function: Creates a NSPanel for the given Xojo window.

Example:

```
MsgBox window1.NSPanelMBS.Title
```

Notes: Works only for Cocoa Windows in Cocoa target which are floating panels.

16.1.4 NSWindowMBS as NSWindowMBS

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

Function: Creates a NSWindow for the given Xojo window.

Example:

```
MsgBox window1.NSWindowMBS.Title
```

Notes: Works only for Cocoa Windows in Cocoa target and for Carbon Windows in carbon targets.

16.2 class Window

16.2.1 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.

16.2.2 Methods

16.2.3 NSPanelMBS as NSPanelMBS

Plugin Version: 12.5, Platform: macOS, Targets: Desktop only.

Function: Creates a NSPanel for the given Xojo window.

Example:

```
MsgBox window1.NSPanelMBS.Title
```

Notes: Works only for Cocoa Windows in Cocoa target which are floating panels.

16.2.4 NSWindowMBS as NSWindowMBS

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

Function: Creates a NSWindow for the given Xojo window.

Example:

```
MsgBox window1.NSWindowMBS.Title
```

Notes: Works only for Cocoa Windows in Cocoa target and for Carbon Windows in carbon targets.

Chapter 17

List of Questions in the FAQ

- 18.0.1 Can anyone help me convert seconds to time in this format hh:mm:ss? 1143
- 18.0.2 Do you have plugins for Android? 1144
- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.6 How to delete a folder? 1147
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150
- 18.0.10 Is there an example for vector graphics in Xojo? 1151
- 18.0.11 Picture functions do not preserve resolution values? 1152
- 18.0.12 A toolbox call needs a rect - how do I give it one? 1152
- 18.0.13 API client not supported? 1152
- 18.0.14 Can I access Access Database with Java classes? 1153
- 18.0.15 Can I create PDF from Xojo Report using DynaPDF? 1154
- 18.0.16 Can I use AppleScripts in a web application? 1154
- 18.0.17 Can I use graphics class with DynaPDF? 1154
- 18.0.18 Can I use sockets on a web application? 1155
- 18.0.19 Can I use your ChartDirector plugin on a web application? 1155

- 18.0.20 Can I use your DynaPDF plugin on a web application? 1156
- 18.0.21 Can I use your plugin controls on a web application? 1157
- 18.0.22 Can you get an unique machine ID? 1157
- 18.0.23 ChartDirector: Alignment Specification 1157
- 18.0.24 ChartDirector: Color Specification 1158
- 18.0.25 ChartDirector: Font Specification 1161
- 18.0.26 ChartDirector: Mark Up Language 1165
- 18.0.27 ChartDirector: Parameter Substitution and Formatting 1169
- 18.0.28 ChartDirector: Shape Specification 1173
- 18.0.29 Copy styled text? 1174
- 18.0.30 Do you have code to validate a credit card number? 1175
- 18.0.31 Do you have plugins for X-Rite EyeOne, eXact or i1Pro? 1176
- 18.0.32 Does SQL Plugin handle stored procedures with multiple result sets? 1176
- 18.0.33 Does the plugin home home? 1176
- 18.0.34 folderitem.absolutePath is limited to 255 chars. How can I get longer ones? 1177
- 18.0.35 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? 1177
- 18.0.36 How about Plugin support for older OS X? 1178
- 18.0.37 How can I detect whether an Intel CPU is a 64bit CPU? 1179
- 18.0.38 How can I disable the close box of a window on Windows? 1180
- 18.0.39 How can I get all the environment variables from Windows? 1180
- 18.0.40 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? 1181
- 18.0.41 How can I get text from a PDF? 1181
- 18.0.42 How can I get text from a Word Document? 1181
- 18.0.43 How can I get the item string for a given file creator? 1182
- 18.0.44 How can I launch an app using it's creator code? 1183
- 18.0.45 How can I learn what shared libraries are required by a plugin on Linux? 1183
- 18.0.46 How can I validate an email address? 1185
- 18.0.47 How do I decode correctly an email subject? 1185

	1135
• 18.0.48 How do I enable/disable a single tab in a tabpanel?	1186
• 18.0.49 How do I find the root volume for a file?	1187
• 18.0.50 How do I get the current languages list?	1187
• 18.0.51 How do I get the Mac OS Version?	1188
• 18.0.52 How do I get the printer name?	1189
• 18.0.53 How do I make a metal window if RB does not allow me this?	1190
• 18.0.54 How do I make a smooth color transition?	1190
• 18.0.55 How do I read the applications in the dock app?	1191
• 18.0.56 How do I truncate a file?	1192
• 18.0.57 How do update a Finder's windows after changing some files?	1192
• 18.0.58 How to access a USB device directly?	1193
• 18.0.59 How to add icon to file on Mac?	1193
• 18.0.60 How to ask the Mac for the Name of the Machine?	1193
• 18.0.61 How to automatically enable retina in my apps?	1194
• 18.0.62 How to avoid leaks with Cocoa functions?	1194
• 18.0.63 How to avoid trouble connecting to oracle database with SQL Plugin?	1195
• 18.0.64 How to avoid ___NSAutoreleaseNoPool console messages in threads?	1195
• 18.0.65 How to bring app to front?	1196
• 18.0.66 How to bring my application to front?	1196
• 18.0.67 How to catch Control-C on Mac or Linux in a console app?	1197
• 18.0.68 How to change name of application menu?	1197
• 18.0.69 How to change the name in the menubar of my app on Mac OS X?	1198
• 18.0.70 How to check if a folder/directory has subfolders?	1198
• 18.0.71 How to check if Macbook runs on battery or AC power?	1199
• 18.0.72 How to check if Microsoft Outlook is installed?	1200
• 18.0.73 How to check on Mac OS which country or language is currently selected?	1200
• 18.0.74 How to code sign my app with plugins?	1201
• 18.0.75 How to collapse a window?	1201
• 18.0.76 How to compare two pictures?	1202

- 18.0.77 How to compile PHP library? 1204
- 18.0.78 How to convert a `BrowserType` to a `String` with `WebSession.Browser`? 1205
- 18.0.79 How to convert a `EngineType` to a `String` with `WebSession.Engine`? 1206
- 18.0.80 How to convert a `PlatformType` to a `String` with `WebSession.Platform`? 1206
- 18.0.81 How to convert a text to iso-8859-1 using the `TextEncoder`? 1207
- 18.0.82 How to convert `ChartTime` back to Xojo date? 1208
- 18.0.83 How to convert line endings in text files? 1208
- 18.0.84 How to convert picture to string and back? 1209
- 18.0.85 How to copy an array? 1210
- 18.0.86 How to copy an dictionary? 1210
- 18.0.87 How to copy parts of a movie to another one? 1210
- 18.0.88 How to create a birthday like calendar event? 1211
- 18.0.89 How to create a GUID? 1212
- 18.0.90 How to create a Mac picture clip file? 1212
- 18.0.91 How to create a PDF file in Xojo? 1213
- 18.0.92 How to create `EmailAttachment` for PDF Data in memory? 1213
- 18.0.93 How to create PDF for image files? 1214
- 18.0.94 How to CURL Options translate to Plugin Calls? 1215
- 18.0.95 How to delete file with ftp and curl plugin? 1216
- 18.0.96 How to detect display resolution changed? 1216
- 18.0.97 How to detect retina? 1217
- 18.0.98 How to disable force quit? 1217
- 18.0.99 How to disable the error dialogs from Internet Explorer on javascript errors? 1217
- 18.0.100 How to display a PDF file in Xojo? 1217
- 18.0.101 How to do a lottery in RB? 1218
- 18.0.102 How to do an asycron DNS lookup? 1219
- 18.0.103 How to draw a dashed pattern line? 1219
- 18.0.104 How to draw a nice antialiased line? 1220
- 18.0.105 How to dump java class interface? 1221

	1137
• 18.0.106 How to duplicate a picture with mask or alpha channel?	1222
• 18.0.107 How to enable assistive devices?	1223
• 18.0.108 How to encrypt a file with Blowfish?	1223
• 18.0.109 How to extract text from HTML?	1224
• 18.0.110 How to find empty folders in a folder?	1224
• 18.0.111 How to find iTunes on a Mac OS X machine fast?	1224
• 18.0.112 How to find network interface for a socket by it's name?	1225
• 18.0.113 How to find version of Microsoft Word?	1226
• 18.0.114 How to fix CURL error 60/53 on connecting to server?	1227
• 18.0.115 How to format double with n digits?	1227
• 18.0.116 How to get a time converted to user time zone in a web app?	1228
• 18.0.117 How to get an handle to the frontmost window on Windows?	1228
• 18.0.118 How to get CFAbsoluteTime from date?	1229
• 18.0.119 How to get client IP address on web app?	1229
• 18.0.120 How to get fonts to load in charts on Linux?	1229
• 18.0.121 How to get fonts to load in DynaPDF on Linux?	1230
• 18.0.122 How to get GMT time and back?	1231
• 18.0.123 How to get good crash reports?	1231
• 18.0.124 How to get list of all threads?	1232
• 18.0.125 How to get parameters from webpage URL in Xojo Web Edition?	1232
• 18.0.126 How to get the color for disabled textcolor?	1232
• 18.0.127 How to get the current free stack space?	1233
• 18.0.128 How to get the current timezone?	1234
• 18.0.129 How to get the current window title?	1235
• 18.0.130 How to get the cursor blink interval time?	1236
• 18.0.131 How to get the list of the current selected files in the Finder?	1237
• 18.0.132 How to get the Mac OS system version?	1238
• 18.0.133 How to get the Mac OS Version using System.Gestalt?	1238
• 18.0.134 How to get the screensize excluding the task bar?	1239

- 18.0.135 How to get the size of the frontmost window on Windows? 1239
- 18.0.136 How to get the source code of a HTMLViewer? 1240
- 18.0.137 How to get Xojo apps running Linux? 1240
- 18.0.138 How to handle really huge images with GraphicsMagick or ImageMagick? 1240
- 18.0.139 How to handle tab key for editable cells in listbox? 1241
- 18.0.140 How to hard link MapKit framework? 1242
- 18.0.141 How to have a PDF downloaded to the user in a web application? 1243
- 18.0.142 How to hide all applications except mine? 1243
- 18.0.143 How to hide script errors in HTMLViewer on Windows? 1244
- 18.0.144 How to hide the grid/background/border in ChartDirector? 1244
- 18.0.145 How to hide the mouse cursor on Mac? 1244
- 18.0.146 How to insert image to NSTextView or TextArea? 1244
- 18.0.147 How to jump to an anchor in a htmlviewer? 1245
- 18.0.148 How to keep a movieplayer unclickable? 1245
- 18.0.149 How to keep my web app from using 100% CPU time? 1246
- 18.0.150 How to kill a process by name? 1246
- 18.0.151 How to know how many CPUs are present? 1247
- 18.0.152 How to know the calling function? 1247
- 18.0.153 How to launch an app using it's creator code? 1248
- 18.0.154 How to launch disc utility? 1248
- 18.0.155 How to make a lot of changes to a REAL SQL Database faster? 1249
- 18.0.156 How to make a NSImage object for my retina enabled app? 1249
- 18.0.157 How to make a window borderless on Windows? 1249
- 18.0.158 How to make an alias using AppleEvents? 1250
- 18.0.159 How to make AppleScripts much faster? 1251
- 18.0.160 How to make double clicks on a canvas? 1251
- 18.0.161 How to make my Mac not sleeping? 1253
- 18.0.162 How to make my own registration code scheme? 1254
- 18.0.163 How to make small controls on Mac OS X? 1254

	1139
• 18.0.164 How to mark my Mac app as background only?	1255
• 18.0.165 How to move a file or folder to trash?	1255
• 18.0.166 How to move an application to the front using the creator code?	1256
• 18.0.167 How to move file with ftp and curl plugin?	1257
• 18.0.168 How to normalize string on Mac?	1257
• 18.0.169 How to obscure the mouse cursor on Mac?	1258
• 18.0.170 How to open icon file on Mac?	1258
• 18.0.171 How to open PDF in acrobat reader?	1258
• 18.0.172 How to open printer preferences on Mac?	1259
• 18.0.173 How to open special characters panel on Mac?	1260
• 18.0.174 How to optimize picture loading in Web Edition?	1260
• 18.0.175 How to parse XML?	1260
• 18.0.176 How to play audio in a web app?	1261
• 18.0.177 How to pretty print xml?	1262
• 18.0.178 How to print to PDF?	1262
• 18.0.179 How to query Spotlight's Last Open Date for a file?	1263
• 18.0.180 How to quit windows?	1264
• 18.0.181 How to read a CSV file correctly?	1264
• 18.0.182 How to read the command line on windows?	1265
• 18.0.183 How to render PDF pages with PDF Kit?	1265
• 18.0.184 How to restart a Mac?	1266
• 18.0.185 How to resume ftp upload with curl plugin?	1266
• 18.0.186 How to rotate a PDF page with CoreGraphics?	1267
• 18.0.187 How to rotate image with CoreImage?	1268
• 18.0.188 How to run a 32 bit application on a 64 bit Linux?	1269
• 18.0.189 How to save HTMLViewer to PDF with landscape orientation?	1269
• 18.0.190 How to save RTFD?	1269
• 18.0.191 How to save RTFD?	1270
• 18.0.192 How to scale a picture proportionally with mask?	1270

- 18.0.193 How to scale a picture proportionally? 1271
- 18.0.194 How to scale/resize a CIImageMBS? 1272
- 18.0.195 How to scale/resize a picture? 1273
- 18.0.196 How to search with regex and use unicode codepoints? 1273
- 18.0.197 How to see if a file is invisible for Mac OS X? 1274
- 18.0.198 How to set cache size for SQLite or REALSQLDatabase? 1275
- 18.0.199 How to set the modified dot in the window? 1275
- 18.0.200 How to show a PDF file to the user in a Web Application? 1275
- 18.0.201 How to show Keyboard Viewer programmatically? 1276
- 18.0.202 How to show the mouse cursor on Mac? 1277
- 18.0.203 How to shutdown a Mac? 1277
- 18.0.204 How to sleep a Mac? 1278
- 18.0.205 How to speed up rasterizer for displaying PDFs with DynaPDF? 1278
- 18.0.206 How to use PDFLib in my RB application? 1278
- 18.0.207 How to use quotes in a string? 1279
- 18.0.208 How to use Sybase in Web App? 1279
- 18.0.209 How to use the Application Support folder? 1279
- 18.0.210 How to use the IOPMCopyScheduledPowerEvents function in Xojo? 1280
- 18.0.211 How to validate a GUID? 1283
- 18.0.212 How to walk a folder hierarchie non recursively? 1283
- 18.0.213 I got this error: PropVal, QDPictMBS.Name (property value), Type mismatch error. Expected CGDataProviderMBS, but got Variant, Name:QDPictMBS 1284
- 18.0.214 I registered the MBS Plugins in my application, but later the registration dialog is shown. 1284
- 18.0.215 I want to accept Drag & Drop from iTunes 1285
- 18.0.216 I'm drawing into a listbox but don't see something. 1287
- 18.0.217 I'm searching for a method or so to move a window from position x.y to somewhere else on the screen. 1287
- 18.0.218 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? 1287
- 18.0.219 Is the fn key on a powerbook keyboard down? 1288

	1141
• 18.0.220 Is there a case sensitive Dictionary?	1288
• 18.0.221 Is there a way to use the MBS plugin to get only the visible item and folder count on a volume?	1289
• 18.0.222 Is there an easy way I can launch the Displays preferences panel?	1289
• 18.0.223 List of Windows Error codes?	1290
• 18.0.224 Midi latency on Windows problem?	1290
• 18.0.225 My Xojo Web App does not launch. Why?	1290
• 18.0.226 SQLiteDatabase not initialized error?	1291
• 18.0.227 Textconverter returns only the first x characters. Why?	1291
• 18.0.228 The type translation between CoreFoundation/Foundation and Xojo data types.	1292
• 18.0.229 Uploaded my web app with FTP, but it does not run on the server!	1294
• 18.0.230 What classes to use for hotkeys?	1294
• 18.0.231 What do I need for Linux to get picture functions working?	1294
• 18.0.232 What does the NAN code mean?	1295
• 18.0.233 What font is used as a 'small font' in typical Mac OS X apps?	1295
• 18.0.234 What is last plugin version to run on Mac OS X 10.4?	1296
• 18.0.235 What is last plugin version to run on PPC?	1296
• 18.0.236 What is last version of the plugins for macOS 32-bit?	1297
• 18.0.237 What is the difference between Timer and WebTimer?	1297
• 18.0.238 What is the list of Excel functions?	1297
• 18.0.239 What is the replacement for PluginMBS?	1298
• 18.0.240 What to do on Xojo reporting a conflict?	1298
• 18.0.241 What to do with a NSImageCacheException?	1299
• 18.0.242 What to do with MySQL Error 2014?	1299
• 18.0.243 What to do with SQL Plugin reporting Malformed string as error?	1299
• 18.0.244 Where is CGGetActiveDisplayListMBS?	1299
• 18.0.245 Where is CGGetDisplaysWithPointMBS?	1300
• 18.0.246 Where is CGGetDisplaysWithRectMBS?	1300
• 18.0.247 Where is CGGetOnlineDisplayListMBS?	1300
• 18.0.248 Where is GetObjectClassNameMBS?	1300

- 18.0.249 Where is NetworkAvailableMBS? 1300
- 18.0.250 Where is StringHeight function in DynaPDF? 1301
- 18.0.251 Where is XLSDocumentMBS class? 1301
- 18.0.252 Where to get information about file formats? 1301
- 18.0.253 Where to register creator code for my application? 1302
- 18.0.254 Which Mac OS X frameworks are 64bit only? 1302
- 18.0.255 Which plugins are 64bit only? 1303
- 18.0.256 Why application doesn't launch because of a missing ddraw.dll!? 1303
- 18.0.257 Why application doesn't launch because of a missing shlwapi.dll!? 1303
- 18.0.258 Why do I hear a beep on keydown? 1303
- 18.0.259 Why does folderitem.item return nil? 1303
- 18.0.260 Why doesn't showurl work? 1303
- 18.0.261 Why don't the picture functions not work on Linux? 1304
- 18.0.262 Why have I no values in my chart? 1304
- 18.0.263 Will application size increase with using plugins? 1304
- 18.0.264 XLS: Custom format string guidelines 1304
- 18.0.265 Xojo doesn't work with your plugins on Windows 98. 1305
- 18.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic. Why? 1306

Chapter 18

The FAQ

18.0.1 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)

18.0.2 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.

18.0.3 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:

- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.6 How to delete a folder? 1147
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.4 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144

- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.6 How to delete a folder? 1147
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.5 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.6 How to delete a folder? 1147
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.6 How to delete a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code that 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.7 How to detect if CPU is 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.6 How to delete a folder? 1147
- 18.0.8 How to query variant type string for a variant? 1149
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.8 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146
- 18.0.6 How to delete a folder? 1147
- 18.0.7 How to detect if CPU is 64bit processor? 1148
- 18.0.9 How to refresh a htmlviewer on Windows? 1150

18.0.9 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:

- 18.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1144
- 18.0.4 How to catch delete key? 1145
- 18.0.5 How to convert cmyk to rgb? 1146

- | | |
|----------------------------------------------------------|------|
| | 1151 |
| • 18.0.6 How to delete a folder? | 1147 |
| • 18.0.7 How to detect if CPU is 64bit processor? | 1148 |
| • 18.0.8 How to query variant type string for a variant? | 1149 |

18.0.10 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
```

18.0.11 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.

18.0.12 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
```

18.0.13 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.

18.0.14 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 library as string = Join(libjs, "")
dim vm as new JavaVMMBS(library)

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>

18.0.15 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.

18.0.16 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.

18.0.17 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: This is a feature request to allow graphics subclasses:

Feedback case 11391: [feedback://showreport?report_id=11391](https://feedback.apple.com/showreport?report_id=11391)

18.0.18 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.

18.0.19 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 chosen 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 &hbbsbbb
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. Transferring 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.

18.0.20 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.

18.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

18.0.22 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.

18.0.23 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.

18.0.24 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 at all. This is equivalent to traditional RGB colors. A non-zero alpha transparency means 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.

A metal color is a color of which the brightness varies smoothly across the chart surface as to make the surface look 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.

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.

A dash line color is a color that switches on and off periodically. When used to draw a line, the line will appear as a dash line.

Dash line colors are created using `BaseChart.dashLineColor` and `DrawArea.dashLineColor`. They accept a line color and a dash pattern code as arguments, and return a 32-bit integer acting as a handle to the dash line color. The handle can be used in any `ChartDirector` API that expects a color as its input.

Zone Colors
A zone color is for XY charts only. It is a color that automatically changes upon reaching a data threshold value along the x-axis or y-axis. Zone colors are created using `Layer.xZoneColor`, `Layer.yZoneColor`, `XYChart.xZoneColor` or `XYChart.yZoneColor`.

Palette Colors
Palette colors are colors of the format `FFFFnnnn`, where the least significant 16 bits (`nnnn`) are the index to the palette. A palette is simply an array of colors. For a palette color, the actual color is obtained by

looking up the palette using the index. For example, the color FFFF0001 is the second color in the palette (first color is index 0).

The colors in the palette can be ARGB colors or "dynamic" colors (pattern, gradient and dash line colors).

The first eight palette colors have special significance. The first three palette colors are the background color, default line color, and default text color of the chart. The 4th to 7th palette colors are reserved for future use. The 8th color is a special dynamic color that is equal to the data color of the "current data set".

The 9th color (index = 8) onwards are used for automatic data colors. For example, in a pie chart, if the sector colors are not specified, ChartDirector will automatically use the 9th color for the first sector, the 10th color for the second sector, and so on. Similarly, for a multi-line chart, if the line colors are not specified, ChartDirector will use the 9th color for the first line, the 10th color for the second line, and so on.

The ChartDirector API defines several constants to facilitate using palette colors.

ConstantValueDescription

Palette	FFFF0000	The starting point of the palette. The first palette color is (Palette + 0). The nth palette color is (Palette + n - 1).
BackgroundColor	FFFF0000	The background color.
LineColor	FFFF0001	The default line color.
TextColor	FFFF0002	The default text color.
[Reserved]	FFFF0003 - FFFF0006	These palette positions are reserved. Future versions of ChartDirector may use these palette positions for colors that have special significance.
SameAsMainColor	FFFF0007	A dynamic color that is equal to the data color of the current data set. This color is useful for objects that are associated with data sets. For example, in a pie chart, if the sector label background color is SameAsMainColor, its color will be the same as the corresponding sector color.
DataColor	FFFF0008	The starting point for the automatic data color allocation.

When a chart is created, it has a default palette. You may modify the palette using BaseChart.setColor, BaseChart.setColors, or BaseChart.setColors2.

The advantages of using palette colors are that you can change the color schemes of the chart in one place. ChartDirector comes with several built-in palettes represented by the following predefined constants.

ConstantDescription

18.0.25 ChartDirector: Font Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

defaultPalette	An array of colors representing the default palette. This palette is designed for drawing charts on white backgrounds (or lightly colored backgrounds).
whiteOnBlackPalette	An array of colors useful for drawing charts on black backgrounds (or darkly colored backgrounds).
transparentPalette	An array of colors useful drawing charts on white backgrounds (or lightly colored backgrounds). The data colors in this palette are all semi-transparent.

Answer: ChartDirector: Font Specification

Notes: Font Name

In ChartDirector, the font name is simply the file name that contains the font. For example, under the Windows platform, the "Arial" font is "arial.ttf", while the "Arial Bold" font is "arialbd.ttf".

NOTE: Mac OS X Specific Information

In Mac OS X, in addition to ".ttf", ChartDirector also supports Mac OS X font file formats, such as Font Suitcase files and Datafork files (.dfont). These files often contain multiple fonts. For example, the "GillSans.dfont" file contains 6 fonts.

So in addition to the file name, an index is needed to determine the font. The index is specified by appending a " | " character to the font name, followed by the index number. For example, the third font in "GillSans.dfont" is denoted as "GillSans.dfont | 2". (Note: The first font starts at 0.) If no index number is provided, the first font is assumed.

ChartDirector also supports using Mac OS X Font Manager names. For example, one may use "Gill Sans Light Italic" instead of using "GillSans.dfont | 1" as the font name. However, the Mac OS X Font Manager is active only if someone has logged into the Mac GUI console, so this method is only recommended for developing applications that run on the GUI console.

The sample programs that come with ChartDirector are designed to run on all operating systems, so they use generic font file names (eg. "arial.ttf") instead of Mac OS X specific names. To allow them to run on Mac OS X, ChartDirector on Mac OS X has a built-in table to map common font file names to Mac OS X font names:

"arial.ttf", "arialbd.ttf", "ariali.ttf" and "arialbi.ttf" are mapped to "Arial | 0" (Arial), "Arial | 1" (Arial Bold), "Arial | 2" (Arial Italic) and "Arial | 3" (Arial Bold Italic)

"times.ttf", "timesbd.ttf", "timesi.ttf" and "timesbi.ttf" are mapped to "Times New Roman | 0" (Times New Roman), "Times New Roman | 1" (Times New Roman Bold), "Times New Roman | 2" (Times New Roman Italic) and "Times New Roman | 3" (Times New Roman Bold Italic)

"cour.ttf", "courbd.ttf", "couri.ttf" and "courbi.ttf" are mapped to "Courier New | 0" (Courier New), "Courier New | 1" (Courier New Bold), "Courier New | 2" (Courier New Italic) and "Courier New | 3" (Courier New Bold Italic)

Font Location

ChartDirector on Windows does not come with any font files. It relies on the operating system's font files in the " [windows] \Fonts" directory. To see what fonts are installed in your operating system and their file names, use the File Explorer to view that directory.

ChartDirector on Windows will also search for the font files in the "fonts" subdirectory (if it exists) under the directory where the ChartDirector DLL "chartdir.dll" is installed. This is useful for private fonts. Also, for some especially secure web servers, the web anonymous user may not have access to the " [windows] \Fonts" directory. In this case, you may copy the font files to the above subdirectory.

ChartDirector on Mac OS X relies on operating system font files in "/Library/Fonts" and "/System/Library/Fonts".

ChartDirector on Linux, FreeBSD and Solaris assume the fonts files are in the "fonts" subdirectory under the directory where the ChartDirector shared object "libchartdir.so" is installed. ChartDirector on Linux, FreeBSD and Solaris come with a number of font files in the "fonts" subdirectory.

To keep the download size small, ChartDirector on Linux, FreeBSD and Solaris only come with some commonly used fonts. You may download additional fonts from the Internet. In particular, the Microsoft fonts at

http://sourceforge.net/project/showfiles.php?group_id=34153&release_id=105355

is highly recommended. Please refer to

<http://www.microsoft.com/typography/faq/faq8.htm>

on how you could use the fonts legally in your system.

ChartDirector supports True Type fonts (.ttf), Type 1 fonts (.pfa and .pfb) and Windows bitmap fonts (.fon). On Mac OS X, ChartDirector also supports Font Suitcase and Datafork (.dfont) files. On Linux, FreeBSD and Solaris, ChartDirector also supports Portable Compiled Fonts (.pcf fonts).

If you want ChartDirector to search other directories for the font files, you may list the directories in an environment variable called "FONTSPATH".

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 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.

18.0.26 ChartDirector: Mark Up Language

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Mark Up Language

Notes: ChartDirector Mark Up Language (CDML) is a language for including formatting information in text strings by marking up the text with tags.

CDML allows a single text string to be rendered using multiple fonts, with different colors, and even embed images in the text. **Font Styles**

You can change the style of the text by using CDML tags. For example, the line:

```
<*font=timesi.ttf,size=16,color=FF0000>Hello <*font=arial.ttf,size=12,color=8000*>world!
```

will result in the following text rendered:

In general, all tags in CDML are enclosed by <*> and *>. Attributes within the tags determine the styles of the text following the tags within the same block.

If you want to include <*> in text without being interpreted as CDML tags, use «* as the escape sequence.

The following table describes the supported font style attributes in CDML. See [Font Specification](#) for details on various font attributes.

AttributeDescription

Set the following text to be in superscript style. This attribute does not need to have a value. (You may use "super" as the attribute instead of "super=1".)

Note that unlike HTML tags, no double or single quotes are used in the tags. It is because CDML tags are often embedded as string literals in source code. The double or single quotes, if used, will conflict with the string literal quotes in the source code. Therefore in CDML, no quotes are necessary and they must not be used.

Also, unlike HTML tags, CDML uses the comma character as the delimiter between attributes. It is because certain attributes may contain embed spaces (such as the font file name). So space is not used as the delimiter and the comma character is used instead.

Note the font attribute above starts a new style section, while other attributes just modify the current style

font	Starts a new style section, and sets the font name. You may use this attribute without a value (that is, use "font" instead of "font=arial.ttf") to create a new style section without modifying the font name.
size	The font size.
width	The font width. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
height	The font height. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
color	The text color in hex format.
bgColor	The background color of the text in hex format.
underline	The line width of the line used to underline the following characters. Set to 0 to disable underline.
sub	Set the following text to be in subscript style. This attribute does not need to have a value. (You may use "sub" as the attribute instead of "sub=1".)
super	Set the following text to be in superscript style.
xoffset	Draw the following the text by shifting the text horizontally from the original position by the specified offset in pixels.
yoffset	Draw the following the text by shifting the text vertically from the original position by the specified offset in pixels.
advance	Move the cursor forward (to the right) by the number of pixels as specified by the value this attribute.
advanceTo	Move the cursor forward (to the right) to the position as specified by the value this attribute. The position is specified as the number of pixels to the right of the left border of the block. If the cursor has already passed through the specified position, the cursor is not moved.

section. You may use `</font*>` to terminate a style section, which will restore the font styles to the state before the style section.

Blocks and Lines

In CDML, a text string may contain multiple blocks. A block may contain multiple lines of text by separating them with new line characters ("`\n`") or with `<br*>`. The latter is useful for programming languages that cannot represent new line characters easily.

For example, the line:

```
<*size=15*><*block*><*color=FF*>BLOCK<*br*>ONE<*/*>and <*block*><*color=FF00*>BLOCK<*br*>TWO
```

will result in the following text rendered:

The above example contains a line of text. The line contains two blocks with the characters " and " in between. Each block in turn contains two lines. The blocks are defined using `<*block*>` as the start tag and

<*/*>as the end tag.

When a block ends, font styles will be restored to the state before entering the block. Embedding Images
CDML supports embedding images in text using the following syntax:

```
<*img=my_image_file.png*>
where my_image_file.png is the path name of the image file.
```

For example, the line:

```
<*size=20*>A <*img=sun.png*>day
will result in the following text rendered:
```

ChartDirector will automatically detect the image file format using the file extension, which must either png, jpg, jpeg, gif, wbmp or wmp (case insensitive).

Please refer to BaseChart.setSearchPath or DrawArea.setSearchPath on the directory that ChartDirector will search for the file.

The <*img*>tag may optionally contain width and height attributes to specify its pixel width and height. In this case, ChartDirector will stretch or compress the image if necessary to the required width and height. Blocks Attributes

CDML supports nesting blocks, that is, a block can contain other sub-blocks. Attributes are supported in the <*block*>tag to control the alignment and orientation of the sub-blocks. The <*img=my_image_file.png*>is treated as a block for layout purposes.

For example, the line:

```
<*block,valign=absmiddle*><*img=molecule.png*><*block*>Hydrazino\nMolecule<*/*><*/*>
will result in the following text rendered:
```

The the above starts <*block,valign=absmiddle*>which specifies its content should align with each others in the vertical direction using the absolute middle alignment. The block contains an image, followed by a space characters, and then another block which has two lines of text.

The following table describes the supported attributes inside <*block*>tag:

AttributeDescription

The value baseline means the baseline of sub-blocks should align with the baseline of the block. The baseline

width	The width of the block in pixels. By default, the width is automatically determined to be the width necessary for the contents of the block. If the width attribute is specified, it will be used as the width of the block. If the width is insufficient for the contents, the contents will be wrapped into multiple lines.
height	The height of the block in pixels. By default, the height is automatically determined to be the height necessary for the contents of the block. If the height attribute is specified, it will be used as the height of the block.
maxwidth	The maximum width of the block in pixels. If the content is wider than maximum width, it will be wrapped into multiple lines.
truncate	The maximum number of lines of the block. If the content requires more than the maximum number of lines, it will be truncated. In particular, if truncate is 1, the content will be truncated if it exceeds the maximum width (as specified by maxwidth or width) without wrapping. The last few characters at the truncation point will be replaced with "...".
linespacing	The spacing between lines as a ratio to the default line spacing. For example, a line spacing of 2 means the line spacing is two times the default line spacing. The default line spacing is the line spacing as specified in the font used.
bgColor	The background color of the block in hex format.
valign	The vertical alignment of sub-blocks. This is for blocks that contain sub-blocks. Supported values are baseline, top, bottom, middle and absmiddle.

is the underline position of text. This is normal method of aligning text, and is the default in CDML. For images or blocks that are rotated, the baseline is the same as the bottom.

The value top means the top line of sub-blocks should align with the top line of the block.

The value bottom means the bottom line of sub-blocks should align with the bottom line of the block.

The value middle means the middle line of sub-blocks should align with the the middle line of the block. The middle line is the middle position between the top line and the baseline.

The value absmiddle means the absolute middle line of sub-blocks should align with the absolute middle line of the block. The absolute middle line is the middle position between the top line and the bottom line.

halign The horizontal alignment of lines. This is for blocks that contain multiple lines. Supported values are left, center and right.

The value left means the left border of each line should align with the left border of the block. This is the default.

The value center means the horizontal center of each line should align with the horizontal center of the block.

The value right means the right border of each line should align with the right border of the block.

angle Rotate the content of the block by an angle. The angle is specified in degrees in counter-clockwise direction.

18.0.27 ChartDirector: Parameter Substitution and Formatting

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Parameter Substitution and Formatting

Notes: ChartDirector charts often contain a lot of text strings. For example, sector labels in pie charts, axis labels for x and y axes, data labels for the data points, HTML image maps, etc, are all text strings.

ChartDirector uses parameter substitution to allow you to configure precisely the information contained in the text and their format.

Format Strings

In parameter substitution, format strings are used to specify the entities to be include into labels and how to format numbers and dates.

For example, when drawing a pie chart with side label layout, the default sector label format string is:

```
" { label } ( { percent } %)"
```

When the sector label is actually drawn, ChartDirector will replace " { label } " with the sector name, and " { percent } " with the sector percentage. So the above label format will result is a sector label similar to "ABC (34.56%)".

You may change the sector label format by changing the format string. For example, you may change it to:

```
" { label } : US$ { value | 2 } K ( { percent } %)"
```

The sector label will then become something like "ABC: US\$ 123.00 (34.56%)".

In general, in ChartDirector parameter substitution, parameters enclosed by curly brackets will be substituted with their actual values when creating the texts.

For parameters that are numbers or dates/times, ChartDirector supports a special syntax in parameter substitution to allow formatting for these values. Please refer to the Number Formatting and Date/Time Formatting sections below for details.

Parameter Expressions

ChartDirector supports numeric expressions in format strings. They are denoted by enclosing the expression with curly brackets and using "=" as the first character. For example:

```
"USD { value } (Euro { = { value } *0.9 } )"
```

In the above, "{ value }" will be substituted with the actual value of the sector. The expression "{ = { value } *0.9 }" will be substituted with the actual value of the sector multiplied by 0.9.

ChartDirector parameter expressions support operators "+", "-", "*", "/", "%" (modulo) and "^" (exponentiation). Operators "*", "/", "%", "^" is computed first, followed by "+" and "-". Operators of the same precedence are computed from left to right). Parenthesis "(" and ")" can be used to change the computation order.

Parameters for Pie Charts

The following table describes the parameters available for pie charts.

Parameter	Description
sector	The sector number. The first sector is 0, while the nth sector is (n-1).
dataSet	Same as { sector } . See above.
label	The text label of the sector.
dataSetName	Same as { label } . See above.
value	The data value of the sector.
percent	The percentage value of the sector.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using BaseChart.addExtraField or BaseChart.addExtraField2.

Parameters for All XY Chart Layers

The followings are parameters that are apply to all XY Chart layers in general. Some layer types may have additional parameters (see below).

Note that certain parameters are inapplicable in some context. For example, when specifying the aggregate label of a stacked bar chart, the { dataSetName } parameter is inapplicable. It is because a stacked bar is composed of multiple data sets. It does not belong to any particular data set and hence does not have a data set name.

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for Line Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Trend Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Box-Whisker Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for HLOC and CandleStick Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Vector Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Parameters for All Polar Layers

The followings are parameters that are apply to all Polar Chart layers in general. Some layer types may have additional parameters (see below).

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for PolarVector Layers

The followings are parameters that are in additional to the parameters for all Polar Chart layers.

Parameters for Axis

The following table describes the parameters available for pie charts.

Number Formatting

For parameters that are numbers, ChartDirector supports a number of formatting options in parameter substitution.

For example, if you want a numeric field { value } to have a precision of two digits to the right of the decimal point, use ',' (comma) as the thousand separator, and use '.' (dot) as the decimal point, and you may use { value | 2, . } . The number 123456.789 will then be displayed as 123,456.79.

For numbers, the formatting options are specified using the following syntax:

```
{ [ param ] | [ a ] [ b ] [ c ] [ d ] }
```

where:

If this field starts with "E" or "e", followed by a number, it means formatting the value using scientific notation with the specified number of decimal places. If the "E" or "e" is not followed by a number, 3 is assumed.

For example, { value | E4 } will format the value 10.3 to 1.0300E+1, and { value | e4 } will format the same value to 1.0300e+1.

If this field starts with "G" or "g", followed by a number, it means formatting the value using the scientific notation only if the value is large and requires more than the specified number of digits, or the value is less than 0.001. If scientific notation is used, the number following "G" or "g" also specifies the number of significant digits to use. If the "G" or "g" is not followed by a number, 4 is assumed.

For example, consider the format string { value | G4 } . The value 10 will be formatted to 10. The value 100000 will be formatted to 1.000E+5. Similarly, for { value | g4 } , the value 10 will be formatted to 10, while the value 100000 will be formatted to 1.000e+5.

If you skip this argument, ChartDirector will display the exact value using at most 6 decimal places.

You may skip [b] [c] [d] . In this case, the default will be used.

Date/Time Formatting

For parameters that are dates/times, the formatting options can be specified using the following syntax:

```
{ [ param ] | [ datetime_format_string ] }
```

where [datetime_format_string] must start with an english character (A-Z or a-z) that is not "G", "g", "E" or "e", and may contain any characters except ' } '. (If it starts with "G", "g", "E" or "e", it will be considered as a number format string.)

Certain characters are substituted according to the following table. Characters that are not substituted will be copied to the output.

For example, a parameter substitution format of { value | mm-dd-yyyy } will display a date as something similar to 09-15-2002. A format of { value | dd/mm/yy hh:nn:ss a } will display a date as something similar to 15/09/02 03:04:05 pm.

If you want to include characters in the format string without substitution, you may enclose the characters in single or double quotes.

For example, the format { value | mmm '<*color=dd0000*>'yyyy } will display a date as something like Jan <*color=dd0000*>2005 (the <*color=dd0000*> is a CDML tag to specify red text color). Note that the <*color=dd0000*> tag is copied directly without substitution, even it contains "dd" which normally will be substituted with the day of month.

Escaping URL/HTML/CDML characters

Parameter substitution is often used to create HTML image maps. In HTML, some characters has special meanings and cannot be used reliably. For example, the '>' is used to represent the end of an HTML tag.

Furthermore, if the field happens to be used as an URL, characters such as '?', '&' and '+' also have special meanings.

By default, ChartDirector will escape template fields used in URL and query parameters when generating image maps. It will modify URL special characters to the URL escape format "%XX" (eg. "?" will become "%3F"). After that, it will modify HTML special characters to the HTML escape format "&#nn;" (eg. ">" will become ">"). Similarly, it will escape other attributes in the image map using HTML escape format (but not URL escape format).

In addition to escaping HTML and URL special characters, ChartDirector will also remove CDML fields in creating image maps. It is because CDML is only interpreted in ChartDirector, should not be useful outside of ChartDirector (such as in browser tool tips).

In some cases, you may not want ChartDirector to escape the special characters. For example, if the parameters have already been escaped before passing to ChartDirector, you may want to disable ChartDirector from escaping them again.

ChartDirector supports the following special fields to control the escape methods - " { escape_url } ", " { noescape_url } ", " { escape_html } ", " { noescape_html } ", " { escape_cdml } " and " { noescape_cdml } ". These fields enable/disable the escape methods used in the template fields that follow them.

18.0.28 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
```

18.0.29 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.

18.0.30 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)

18.0.31 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.

Then we can talk about the plugin.

18.0.32 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. Then 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.

18.0.33 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.

18.0.34 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
```

18.0.35 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.

18.0.36 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.

18.0.37 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.

18.0.38 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.

18.0.39 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.

18.0.40 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.

18.0.41 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.

18.0.42 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.

18.0.43 How can I get the item string for a given file creator?

Plugin Version: all.

Answer: Try this function:

Example:

```
Sub pullNativeDocs(aCREA As string)
Dim result as Integer
Dim m, k as memoryBlock
Dim f as folderItem
Dim newType as string
Dim anIcon As picture
Dim ofs as Integer
```

```
Declare Function GetFileTypesThatAppCanNativelyOpen Lib "Carbon" (appVRefNumHint as Short, appSignature as OSType, nativeTypes as Ptr) as Short Inline68K("701CABFC")
Declare Function GetDocumentKindString Lib "Carbon" (docVRefNum as Short, docType as OSType, docCreator as OSType, kindString as ptr) as Short Inline68K("7016ABFC")
```

```
listBox1.deleteAllRows
```

```
m = newMemoryBlock(1024)
result = GetFileTypesThatAppCanNativelyOpen(Volume(0).MacVRefNum, aCREA, m)
if result <> 0 then
listBox1.addRow "<Not found.>"
return
end if
```

```
do
if m.byte(ofs*4) = 0 then
exit
else
newType = m.OSTypeMBS(ofs*4)
listBox1.addRow newType
k = newMemoryBlock(64)
result = GetDocumentKindString(Volume(0).MacVRefNum, newType, aCREA, k)
if result = 0 then
listBox1.cell(ofs,1) = k.pString(0)
ofs = ofs + 1
else
listBox1.cell(ofs,1) = "(unknown)"
end if
end if
```

loop

End Sub

Notes: Change "Translation" to "CarbonLib" for Mac OS X.

18.0.44 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
```

18.0.45 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 name. If a library is missing, you usually see the address missing there or being zero.


```

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.

18.0.48 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.

18.0.49 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
```

18.0.50 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.

18.0.51 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.

18.0.52 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.

18.0.53 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.

18.0.54 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.

18.0.55 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 CObjectMBS = 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.

18.0.56 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.

18.0.57 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.

18.0.58 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.

18.0.59 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.

18.0.60 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).

18.0.61 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.

18.0.62 How to avoid leaks with Cocoa functions?

Plugin Version: all, Platform: macOS.

Answer: You can try this code on Mac OS X:

Example:

```

// in a Timer Action event:
Sub Action()
static LastPool as NSAutoreleasePoolMBS = nil
static CurrentPool as NSAutoreleasePoolMBS = nil

```

```

LastPool = CurrentPool
CurrentPool = new NSAutoreleasePoolMBS

```

End Sub

Notes: With Xojo 2009r4 the code above should not be needed as Xojo runtime does automatically handle the `NSAutoreleasePools` for you. For older Xojo versions you need to use code with a timer with the action event above to avoid memory leaks.

Please do not use Xojo 2009r4 and newer with plugins before version 9.5. You can get crashes there which typically show a line with a `objc_msgSend` call.

18.0.63 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>
```

18.0.64 How to avoid `___NSAutoreleaseNoPool` console messages in threads?

Plugin Version: all, Platform: macOS.

Answer: You need to use your own `NSAutoreleasePool` on a thread like this:

Example:

```
sub MyThread.run
dim pool as new NSAutoreleasePoolMBS
// do work here

pool=nil
```

end sub

Notes: For more details read here:

http://developer.apple.com/mac/library/documentation/Cocoa/Reference/Foundation/Classes/NSAutoreleasePool_Class/Reference/Reference.html

18.0.65 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.

18.0.66 How to bring my application to front?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code txtxt) 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)

18.0.67 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 caught, a flag is set and you can ask later in your normal application flow for the result.

18.0.68 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.

18.0.69 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/>.

18.0.70 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.

18.0.71 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.

18.0.72 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
```

18.0.73 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.

18.0.74 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.

18.0.75 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.

18.0.76 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.

18.0.77 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.

18.0.78 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
```

18.0.79 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
```

18.0.80 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

18.0.81 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 although 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.

18.0.82 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.

18.0.83 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.

18.0.84 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

18.0.85 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.

18.0.86 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.

18.0.87 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.

18.0.88 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.

18.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

18.0.90 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("AQAAAAAAAAAAAAAAAAACAFRDRVIAAABAAAAAAAAAAABUQ0IQAAAAA")
r.AddResource(data,"drag",128,"") 'ditto
r.Close
```

Notes: In general Apple has deprecated this, but a few application still support clippings.

18.0.91 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.

18.0.92 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.

18.0.93 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.

18.0.94 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 interpreted 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. Pass this additional header with the `SetOptionHTTPHeader` method.

```
curl -X PUT http://127.0.0.1:5984/appserials/f2f4e540bf8bb60f61cfc4328001c59 -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)

18.0.95 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.

18.0.96 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".

18.0.97 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)
```

18.0.98 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.

18.0.99 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.

18.0.100 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.

18.0.101 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(5))+chr(13)+str(za(6))+chr(13)+str(za(7))+chr(13)+str(za(8))+chr(13)+str(za(9))+chr(13)+str(za(10))+chr(13)+str(za(11))+chr(13)+str(za(12))+chr(13)+str(za(13))+chr(13)+str(za(14))+chr(13)+str(za(15))+chr(13)+str(za(16))+chr(13)+str(za(17))+chr(13)+str(za(18))+chr(13)+str(za(19))+chr(13)+str(za(20))+chr(13)+str(za(21))+chr(13)+str(za(22))+chr(13)+str(za(23))+chr(13)+str(za(24))+chr(13)+str(za(25))+chr(13)+str(za(26))+chr(13)+str(za(27))+chr(13)+str(za(28))+chr(13)+str(za(29))+chr(13)+str(za(30))+chr(13)+str(za(31))+chr(13)+str(za(32))+chr(13)+str(za(33))+chr(13)+str(za(34))+chr(13)+str(za(35))+chr(13)+str(za(36))+chr(13)+str(za(37))+chr(13)+str(za(38))+chr(13)+str(za(39))+chr(13)+str(za(40))+chr(13)+str(za(41))+chr(13)+str(za(42))+chr(13)+str(za(43))+chr(13)+str(za(44))+chr(13)+str(za(45))+chr(13)+str(za(46))+chr(13)+str(za(47))+chr(13)+str(za(48))+chr(13)+str(za(49))+chr(13)+str(za(50))+chr(13)+str(za(51))+chr(13)+str(za(52))+chr(13)+str(za(53))+chr(13)+str(za(54))+chr(13)+str(za(55))+chr(13)+str(za(56))+chr(13)+str(za(57))+chr(13)+str(za(58))+chr(13)+str(za(59))+chr(13)+str(za(60))+chr(13)+str(za(61))+chr(13)+str(za(62))+chr(13)+str(za(63))+chr(13)+str(za(64))+chr(13)+str(za(65))+chr(13)+str(za(66))+chr(13)+str(za(67))+chr(13)+str(za(68))+chr(13)+str(za(69))+chr(13)+str(za(70))+chr(13)+str(za(71))+chr(13)+str(za(72))+chr(13)+str(za(73))+chr(13)+str(za(74))+chr(13)+str(za(75))+chr(13)+str(za(76))+chr(13)+str(za(77))+chr(13)+str(za(78))+chr(13)+str(za(79))+chr(13)+str(za(80))+chr(13)+str(za(81))+chr(13)+str(za(82))+chr(13)+str(za(83))+chr(13)+str(za(84))+chr(13)+str(za(85))+chr(13)+str(za(86))+chr(13)+str(za(87))+chr(13)+str(za(88))+chr(13)+str(za(89))+chr(13)+str(za(90))+chr(13)+str(za(91))+chr(13)+str(za(92))+chr(13)+str(za(93))+chr(13)+str(za(94))+chr(13)+str(za(95))+chr(13)+str(za(96))+chr(13)+str(za(97))+chr(13)+str(za(98))+chr(13)+str(za(99))
End Sub

```

18.0.102 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 asycron.

18.0.103 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.

18.0.104 How to draw a nice antialiased line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you although 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.

18.0.105 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

18.0.106 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.

18.0.107 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.

18.0.108 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.

18.0.109 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: Grüë

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 ü.

18.0.110 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
```

18.0.111 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
```

18.0.112 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.

18.0.113 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".

18.0.114 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/>

18.0.115 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.

18.0.116 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
```

18.0.117 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
```

18.0.118 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 CFGregorianCalendarMBS
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.

18.0.119 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
```

18.0.120 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.

18.0.121 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.

18.0.122 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.

18.0.123 How to get good crash reports?

Plugin Versions: all, Platforms: macOS, Linux, Windows.

Answer: Check this website from the webkit website:

Notes: <http://webkit.org/quality/crashlogs.html>

18.0.124 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.

18.0.125 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.

18.0.126 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" ...

18.0.127 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 ThreadCurrentStackSize lib "Carbon" (threadid as Integer, byref size as Integer) as short
```

```
if GetCurrentThread(threadid)=0 then
```

```
if 0=ThreadCurrentStackSize(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.

18.0.128 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

```

18.0.129 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 Accessibility 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

```

18.0.130 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.

18.0.131 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.folderItem(i)

dateiname=f.name
// editfield1 with property "multiline=true"!
editfield1.text=editfield1.text + dateiname + chr(13)
next
```

end if

18.0.132 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

```

18.0.133 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.

18.0.134 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.

18.0.135 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!

18.0.136 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
```

18.0.137 How to get Xojo apps running Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You need to install some require 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.

18.0.138 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.

18.0.139 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.

18.0.140 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.

18.0.141 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.

18.0.142 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
```

18.0.143 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).

18.0.144 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.

18.0.145 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.

18.0.146 How to insert image to NSTextView or TextArea?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to insert file:

Example:

```
// insert a file to textview
```

```
Public Sub InsertFile(textview as NSTextViewMBS, f as FolderItem)
```

```
// read to file
```

```

dim b as BinaryStream = BinaryStream.Open(f)
dim s as string = b.Read(b.Length)

// build wrapper
dim fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
dim fileAttachment as new NSTextAttachmentMBS(fileWrapper)
dim attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub

```

Notes: For TextArea you can query the underlying NSTextViewMBS object via TextArea.NSTextViewMBS method.

18.0.147 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

```

18.0.148 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
```

18.0.149 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.

18.0.150 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.

18.0.151 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.

18.0.152 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.

18.0.153 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
```

18.0.154 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.

18.0.155 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.

18.0.156 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.

18.0.157 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 = &H8080000
```

```
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
```

18.0.158 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

18.0.159 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

```

18.0.160 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 GetDbtTime Lib "Carbon" () as Integer
doubleClickTime = GetDbtTime()
#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 occurred close enough together in space
if abs(X - lastClickX) <= 5 and abs(Y - LastClickY) <= 5 then
DoubleClick //a double click has occurred 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

18.0.161 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 HDAActivity = 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.

18.0.162 How to make my own registration code scheme?

Plugin Version: all, Platform: Windows.

Answer: There are excellent articles about how to make a registration 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?

18.0.163 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))

```

18.0.164 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.

18.0.165 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.

18.0.166 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)

18.0.167 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 "RNT0 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 RNT0 with the new file name. To delete use DELE and the file path.

18.0.168 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.

18.0.169 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.

18.0.170 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
```

18.0.171 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.

18.0.172 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

```

18.0.173 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.

18.0.174 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.

18.0.175 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.

18.0.176 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 javascript.

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 ele-
ment.</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)+"");")
```

18.0.177 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>

18.0.178 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.

18.0.179 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.

18.0.180 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.

18.0.181 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.

18.0.182 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.

18.0.183 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.

18.0.184 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

```

18.0.185 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.

18.0.186 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.

18.0.187 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
```

18.0.188 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.

18.0.189 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.

18.0.190 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.rtf")

if f = nil then Return

dim a as NSAttributedStringMBS = textView.textStorage
dim w as NSFFileWrapperMBS = 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 underlying NSTextViewMBS object via TextArea.NSTextViewMBS method.

18.0.191 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.

18.0.192 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 ProportionalScaledWithMask(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.

18.0.193 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)

18.0.194 How to scale/resize a CIIImageMBS?

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 CIIImageMBS = CIIImageMBS.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.

18.0.195 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.

18.0.196 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 √√√° 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

```

18.0.197 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

```

18.0.198 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.SQLiteExecute "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.

18.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:

Example:

```

window1.ModifiedMBS=true

```

18.0.200 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.

18.0.201 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
```

18.0.202 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.

18.0.203 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 immediatly: No document asked to be saved, all applications quitting without knowing).

```
Declare Sub ShutDwnPower Lib "Carbon" ()
ShutDwnPower
```

18.0.204 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
```

18.0.205 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.

18.0.206 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.

18.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.

Example:

```
msgbox "This String contains ""quotes"""
```

18.0.208 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
```

18.0.209 How to use the Application Support folder?

Plugin Version: all, Platform: macOS.

Answer:

I was saving a registration code for an app to the Preference folder. 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!

18.0.210 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 carrayMBS
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 CFDictionaryes
'* 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.

18.0.211 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".

18.0.212 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.

18.0.213 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.

18.0.214 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 temporary 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.

18.0.215 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(NewCFBinaryDataMBS(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=NewCFURLMBS(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.

18.0.216 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 clipping and an offset which the plugin doesn't know about.

18.0.217 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.

18.0.218 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)

18.0.219 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.

18.0.220 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
```

18.0.221 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.

18.0.222 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
```

18.0.223 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>

18.0.224 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)

18.0.225 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? Then 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?

18.0.226 SQLiteDatabase not initialized error?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Before you can use SQLiteDatabaseMBS, it must be initialized.

Example:

```
dim d as new SQLiteDatabaseMBS
```

Notes: This happens normally when you use "new SQLiteDatabaseMBS".

But if you just have a SQLConnectionMBS and get a recordset there, the initialization may not have happened, yet.

So please simply add a line "dim d as new SQLiteDatabaseMBS" to your app.open code after registration, so the plugin part can initialize and late provide recordsets.

18.0.227 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.

18.0.228 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 ->NSData

Carbon CTypeRef 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 CTypeRef:

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.

18.0.229 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.

18.0.230 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.

18.0.231 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.

18.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

18.0.233 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
fontName 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

```

18.0.234 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.

18.0.235 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.

18.0.236 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 use older versions if you use an old Xojo.

Xojo 2017r3 and newer load our 64-bit plugins.

18.0.237 What is the difference between Timer and WebTimer?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Timer is server side and WebTimer client side.

Notes: Timer is the normal timer class in Xojo. It runs on the server. On the client 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.

18.0.238 What is the list of Excel functions?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below is a list of function names known by LibXL.

Notes: LibXL parses the functions and writes tokens to the excel file. So even if Excel can do more functions, we can only accept the ones known by LibXL.

ABS, ABSREF, ACOS, ACOSH, ACTIVE.CELL, ADD.BAR, ADD.COMMAND, ADD.MENU, ADD.TOOLBAR, ADDRESS, AND, APP.TITLE, AREAS, ARGUMENT, ASC, ASIN, ASINH, ATAN, ATAN2, ATANH, AVEDEV, AVERAGE, AVERAGEA, BAHTTEXT, BETADIST, BETAINV, BINOMDIST, BREAK, CALL, CALLER, CANCEL.KEY, CEILING, CELL, CHAR, CHECK.COMMAND, CHIDIST, CHIINV, CHITEST, CHOOSE, CLEAN, CODE, COLUMN, COLUMNS, COMBIN, CONCATENATE, CONFIDENCE, CORREL, COS, COSH, COUNT, COUNTA, COUNTBLANK, COUNTIF, COVAR, CREATE.OBJECT, CRITBINOM, CUSTOM.REPEAT, CUSTOM.UNDO, DATE, DATEDIF, DATESTRING, DATEVALUE, DAVERAGE, DAY, DAYS360, DB, DBCS, DCOUNT, DCOUNTA, DDB, DEGREES, DELETE.BAR, DELETE.COMMAND, DELETE.MENU, DELETE.TOOLBAR, Deref, DEVSQ, DGET, DIALOG.BOX, DIRECTORY, DMAX, DMIN, DOCUMENTS, DOLLAR, DPRODUCT, DSTDEV, DSTDEVP, DSUM, DVAR, DVARP, ECHO, ELSE, ELSE.IF, ENABLE.COMMAND, ENABLE.TOOL, END.IF, ERROR, ERROR.TYPE, EVALUATE, EVEN, EXACT, EXEC, EXECUTE, EXP, EXPONDIST, FACT, FALSE, FCLOSE, FDIST, FILES, FIND, FINDB, FINV, FISHER, FISHERINV, FIXED, FLOOR, FOPEN, FOR, FOR.CELL, FORECAST,

FORMULA.CONVERT, FPOS, FREAD, FREADLN, FREQUENCY, FSIZE, FTEST, FV, FWRITE, FWRITELN, GAMMADIST, GAMMAINV, GAMMALN, GEOMEAN, GET.BAR, GET.CELL, GET.CHART.ITEM, GET.DEF, GET.DOCUMENT, GET.FORMULA, GET.LINK.INFO, GET.MOVIE, GET.NAME, GET.NOTE, GET.OBJECT, GET.PIVOT.FIELD, GET.PIVOT.ITEM, GET.PIVOT.TABLE, GET.TOOL, GET.TOOLBAR, GET.WINDOW, GET.WORKBOOK, GET.WORKSPACE, GETPIVOTDATA, GOTO, GROUP, GROWTH, HALT, HARMEAN, HELP, HLOOKUP, HOUR, HYPERLINK, HYPGEOMDIST, IF, INDEX, INDIRECT, INFO, INITIATE, INPUT, INT, INTERCEPT, IPMT, IRR, ISBLANK, ISERR, ISERROR, ISLOGICAL, ISNA, ISNONTEXT, ISNUMBER, ISPMT, ISREF, ISTEXT, ISTHAIDIGIT, KURT, LARGE, LAST.ERROR, LEFT, LEFTB, LEN, LENB, LINEST, LINKS, LN, LOG, LOG10, LOGEST, LOGINV, LOGNORMDIST, LOOKUP, LOWER, MATCH, MAX, MAXA, MDETERM, MEDIAN, MID, MIDB, MIN, MINA, MINUTE, MINVERSE, MIRR, MMULT, MOD, MODE, MONTH, MOVIE.COMMAND, N, NA, NAMES, NEGBINOMDIST, NEXT, NORMDIST, NORMINV, NORMSDIST, NORMSINV, NOT, NOTE, NOW, NPER, NPV, NUMBERSTRING, ODD, OFFSET, OPEN.DIALOG, OPTIONS.LISTS.GET, OR, PAUSE, PEARSON, PERCENTILE, PERCENTRANK, PERMUT, PHONETIC, PI, PIVOT.ADD.DATA, PMT, POISSON, POKE, POWER, PPMT, PRESS.TOOL, PROB, PRODUCT, PROPER, PV, QUARTILE, RADIANS, RAND, RANK, RATE, REFTTEXT, REGISTER, REGISTER.ID, RELREF, RENAME.COMMAND, REPLACE, REPLACEB, REPT, REQUEST, RESET.TOOLBAR, RESTART, RESULT, RESUME, RETURN, RIGHT, RIGHTB, ROMAN, ROUND, ROUNDBAHTDOWN, ROUNDBAHTUP, ROUNDDOWN, ROUNDUP, ROW, ROWS, RSQ, RTD, SAVE.DIALOG, SAVE.TOOLBAR, SCENARIO.GET, SEARCH, SEARCHB, SECOND, SELECTION, SERIES, SET.NAME, SET.VALUE, SHOW.BAR, SIGN, SIN, SINH, SKEW, SLN, SLOPE, SMALL, SPELLING.CHECK, SQRT, STANDARDIZE, STDEV, STDEVA, STDEVP, STDEVPA, STEP, STEYX, SUBSTITUTE, SUBTOTAL, SUM, SUMIF, SUMPRODUCT, SUMSQ, SUMX2MY2, SUMX2PY2, SUMXMY2, SYD, T, TAN, TANH, TDIST, TERMINATE, TEXT, TEXT.BOX, TEXTREF, THAIDAYOFWEEK, THAIDIGIT, THAIMONTHOFYEAR, THAINUMSOUND, THAINUMSTRING, THAISTRINGLENGTH, THAIYEAR, TIME, TIMEVALUE, TINV, TODAY, TRANSPOSE, TREND, TRIM, TRIMMEAN, TRUE, TRUNC, TTEST, TYPE, UNREGISTER, UPPER, USDOLLAR, USERDEFINED, VALUE, VAR, VARA, VARP, VARPA, VDB, VIEW.GET, VLOOKUP, VOLATILE, WEEKDAY, WEIBULL, WHILE, WINDOW.TITLE, WINDOWS, YEAR and ZTEST.

18.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

18.0.240 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.

18.0.241 What to do with a NSImageCacheException?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You need to add exception handlers for NSExcptionMBS 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.

18.0.242 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.

18.0.243 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.

18.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

18.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

18.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

18.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

18.0.248 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.

18.0.249 Where is NetworkAvailableMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We removed NetworkAvailableMBS some versions ago. It was not working right and basically it's not useful. If you want to check whether you have a network, than do a DNS resolve:

Example:

```

// two independent 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.

18.0.250 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.

18.0.251 Where is XLSDocumentMBS class?

Plugin Version: all, Platform: macOS.

Answer: This class has been removed in favor of XLBookMBS class.

Notes: This classes have been removed XLSCellMBS, XLSDocumentMBS, XLSFormatRecordMBS, XLSMergedCellsMBS, XLSRowMBS and XLSSheetMBS.

18.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:

<http://www.wotsit.org>

18.0.253 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>

18.0.254 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.

18.0.255 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.

18.0.256 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.

18.0.257 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.

18.0.258 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.

18.0.259 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.

18.0.260 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.

18.0.261 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.

18.0.262 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.

18.0.263 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.

18.0.264 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

18.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

18.0.266 Xoho 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 preferred 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 <code>Axis.setLabels</code> 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 <code>Layer.addExtraField</code> , <code>Layer.addExtraField2</code> , <code>BaseChart.addExtraField</code> or <code>BaseChart.addExtraField2</code> .

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 <code>LineStyle.setSymbolScale</code> .
zy	The symbol scale in the y dimension. Applicable for layers with symbol scales set by <code>LineStyle.setSymbolScale</code> .
z	The symbol scale without distinguishing the dimension to use. Applicable for layers with symbol scales set by <code>LineStyle.setSymbolScale</code> .

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 Polar-Layer.setSymbolScale.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using Layer.addExtraField, Layer.addExtraField2, BaseChart.addExtraField or BaseChart.addExtraField2.

diFieldN	Same as fieldN. See above.
dsFieldN	Similar to fieldN, except that dsFieldN means the extra field is indexed by layer index. The Pth layer corresponds to the Pth element of the extra field.
dsdiFieldN	Similar to fieldN, except that dsdiFieldN means the extra fields are indexed by both the data set number and data point number. The Pth data item of the Qth layer corresponds to the Pth element of the (N + Q)th extra field.

Parameter	Description
dir	The direction of the vector.
len	The length of the vector.

Parameter	Description
value	The axis value at the tick position.
label	The axis label at the tick position.

Parameter	Description
[param]	The name of the parameter
[a]	If this field a number, it specifies the number of decimal places (digits to the right of the decimal point).

[b]	The thousand separator. Should be a non-alphanumeric character (not 0-9, A-Z, a-z). Use ' '.
textasciitilde ' for no thousand separator. The default is ' '.	
textasciitilde ', which can be modified using BaseChart.setNumberFormat.	
[c]	The decimal point character. The default is '.', which can be modified using BaseChart.setNumberFormat.
[d]	The negative sign character. Use ' '.
textasciitilde ' for no negative sign character. The default is '-', which can be modified using BaseChart.setNumberFormat.	

Parameter	Description
yyyy	The year in 4 digits (e.g. 2002)
yyy	The year showing only the least significant 3 digits (e.g. 002 for the year 2002)
yy	The year showing only the least significant 2 digits (e.g. 02 for the year 2002)
y	The year showing only the least significant 1 digits (e.g. 2 for the year 2002)
mmm	The month formatted as its name. The default is to use the first 3 characters of the english month name (Jan, Feb, Mar ...). The names can be configured using BaseChart.setMonthNames.
mm	The month formatted as 2 digits from 01 - 12, adding leading zero if necessary.
m	The month formatted using the minimum number of digits from 1 - 12.
MMM	The first 3 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
MM	The first 2 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
M	The first character of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
dd	The day of month formatted as 2 digits from 01 - 31, adding leading zero if necessary.
d	The day of month formatted using the minimum number of digits from 1 - 31.
w	The name of the day of week. The default is to use the first 3 characters of the english day of week name (Sun, Mon, Tue ...). The names can be configured using BaseChart.setWeekDayNames.
hh	The hour of day formatted as 2 digits, adding leading zero if necessary. The 2 digits will be 00 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
h	The hour of day formatted using the minimum number of digits. The digits will be 0 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
nn	The minute formatted as 2 digits from 00 - 59, adding leading zero if necessary.
n	The minute formatted using the minimum number of digits from 00 - 59.
ss	The second formatted as 2 digits from 00 - 59, adding leading zero if necessary.
s	The second formatted using the minimum number of digits from 00 - 59.
a	Display either 'am' or 'pm', depending on whether the time is in the morning or afternoon. The text 'am' and 'pm' can be modified using BaseChart.setAMPM.

Shape Id	Value	Description
SquareShape	1	Square shape. See (1, 1) above.
DiamondShape	2	Diamond shape. See (2, 1) above.
TriangleShape	3	Triangle shape pointing upwards. See (3, 1) above.
RightTriangleShape	4	Triangle shape pointing rightwards. See (4, 1) above.
LeftTriangleShape	5	Triangle shape pointing leftwards. See (5, 1) above.
InvertedTriangleShape	6	Triangle shape pointing downwards. See (1, 2) above.
CircleShape	7	Circle shape. See (2, 2) above.
StarShape	[Method]	Star shapes of various points. See (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5) above for stars with 3 to 10 points.
PolygonShape	[Method]	Polygon shapes symmetrical about a vertical axis with a vertex at the top center position. See (4, 1), (4, 3), (4, 5), (5, 1) for polygons of 5 to 8 sides.
Polygon2Shape	[Method]	Polygon shapes symmetrical about a vertical axis but without any vertex at the top center position. See (4, 2), (4, 4) for polygons of 5 and 6 sides.
CrossShape	[Method]	'+' shapes. See (5, 2), (5, 3), (5, 4), (5, 5), (6, 1), (6, 2), (6, 3) for '+' shape with arm width of 0.1 - 0.7.
Cross2Shape	[Method]	'X' shapes. See (6, 4), (6, 5), (7, 1), (7, 2), (7, 3), (7, 4), (7, 5) for 'X' shapes with arm width of 0.1 - 0.7.

langEnglish	0	Roman script
langFrench	1	Roman script
langGerman	2	Roman script
langItalian	3	Roman script
langDutch	4	Roman script
langSwedish	5	Roman script
langSpanish	6	Roman script
langDanish	7	Roman script
langPortuguese	8	Roman script
langNorwegian	9	Roman script
langHebrew	10	Hebrew script
langJapanese	11	Japanese script
langArabic	12	Arabic script
langFinnish	13	Roman script
langGreek	14	Greek script using smRoman script code
langIcelandic	15	modified smRoman/Icelandic script
langMaltese	16	Roman script
langTurkish	17	modified smRoman/Turkish script
langCroatian	18	modified smRoman/Croatian script
langTradChinese	19	Chinese (Mandarin) in traditional characters
langUrdu	20	Arabic script
langHindi	21	Devanagari script
langThai	22	Thai script
langKorean	23	Korean script

Nan	Meaning
1	Invalid square root (negative number, usually)
2	Invalid addition (indeterminate such as infinity + (-infinity))
4	Invalid division (indeterminate such as 0/0)
8	Invalid multiplication (indeterminate such as 0*infinity)
9	Invalid modulo such as (a mod 0)
17	Try to convert invalid string to a number like val("x7")
33	Invalid argument in a trig function
34	Invalid argument in an inverse trig function
36	Invalid argument in a log function
37	Invalid argument in Pow function
38	Invalid argument in toolbox financial function
40	Invalid argument in hyperbolic function
42	Invalid argument in a gamma function

Symbol	Description and result
0	Digit placeholder. For example, if the value 8.9 is to be displayed as 8.90, use the format #.00
#	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall not display extra zeros when the number typed has fewer digits on either side of the decimal than there are # symbols in the format. For example, if the custom format is #.##, and 8.9 is in the cell, the number 8.9 is displayed.
?	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall put a space for insignificant zeros on either side of the decimal point so that decimal points are aligned in the column. For example, the custom format 0.0? aligns the decimal points for the numbers 8.9 and 88.99 in a column.
. (period)	Decimal point.
%	Percentage. If the cell contains a number between 0 and 1, and the custom format 0% is used, the application shall multiply the number by 100 and add 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,,